AN EFFICIENT BAYESIAN FORMULATION FOR PRODUCTION DATA INTEGRATION INTO RESERVOIR MODELS

A Dissertation

by

LEONARDO VEGA VELASQUEZ

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

December 2003

Major Subject: Petroleum Engineering

AN EFFICIENT BAYESIAN FORMULATION FOR PRODUCTION DATA INTEGRATION INTO RESERVOIR MODELS

A Dissertation

by

LEONARDO VEGA VELASQUEZ

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved as to style and content by:

Akhil Datta-Gupta (Chair of Committee)	Steve A. Holditch (Member)
W. John Lee (Member)	Bani K. Mallick (Member)
Hans Juvkam-Wold (Head of Department)	

December 2003

Major Subject: Petroleum Engineering

ABSTRACT

An Efficient Bayesian Formulation for Production Data Integration into Reservoir Models. (December 2003)

Leonardo Vega Velásquez, B.S., Universidad Nacional de Colombia;

M.S., Texas A&M University

Chair of Advisory Committee: Dr. Akhil Datta-Gupta

Current techniques for production data integration into reservoir models can be broadly grouped into two categories: deterministic and Bayesian. The deterministic approach relies on imposing parameter smoothness constraints using spatial derivatives to ensure large-scale changes consistent with the low resolution of the production data. The Bayesian approach is based on prior estimates of model statistics such as parameter covariance and data errors and attempts to generate posterior models consistent with the static and dynamic data. Both approaches have been successful for field-scale applications although the computational costs associated with the two methods can vary widely. This is particularly the case for the Bayesian approach that utilizes a prior covariance matrix that can be large and full. To date, no systematic study has been carried out to examine the scaling properties and relative merits of the methods.

The main purpose of this work is twofold. First, we systematically investigate the scaling of the computational costs for the deterministic and the Bayesian approaches for realistic field-scale applications. Our results indicate that the deterministic approach exhibits a linear increase in the CPU time with model size compared to a quadratic increase for the Bayesian approach. Second, we propose a fast and robust adaptation of the Bayesian formulation that preserves the statistical foundation of the Bayesian method and at the same time has a scaling property similar to that of the deterministic approach. This can lead to orders of magnitude savings in computation time for model sizes greater than 100,000 grid blocks. We demonstrate the power and utility of our proposed method

using synthetic examples and a field example from the Goldsmith field, a carbonate reservoir in west Texas.

The use of the new efficient Bayesian formulation along with the Randomized Maximum Likelihood method allows straightforward assessment of uncertainty. The former provides computational efficiency and the latter avoids rejection of expensive conditioned realizations.

DEDICATION

To God…to life…to those brief moments of inspiration…and those long moments of perspiration

ACKNOWLEDGMENTS

To Dr. Wattenbarger for the opportunity to begin,

To Darla and Dr. Lee for teaching me how to speak,

To Dr. Mamora and Dr. Holditch for keeping me afloat,

To Dr. Datta-Gupta for keeping me challenged and being a good friend.

TABLE OF CONTENTS

CHAPTER Page

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

CHAPTER I

INTRODUCTION

The practice of inferring reservoir property distributions from dynamic observations of reservoir performance such as transient pressure/tracer response or production data typically involves the solution of an inverse problem.[1-10] Such inverse problems for reservoir characterization are typically undetermined and can lead to instability and non-uniqueness in the solution.[11-12] To remedy the situation, we generally resort to data-independent prior information that can limit the 'plausible' models that satisfy the data. I will examine two different approaches for incorporating prior information during production data integration into reservoir models: 'Bayesian' and 'deterministic'.[11, 13-15] The two approaches differ fundamentally in the way in which probability is introduced into the calculation and their treatment of observed data and prior information.[13] The Bayesian approach associates probability with the prior information whereas the deterministic approach treats it as fixed. In fact, in the deterministic approach probability enters into the calculation only via the data errors which generally have a random component associated with it.

My goal in this dissertation is not to advocate either the Bayesian or the deterministic approach during production data integration. Both approaches have been used very successfully under a wide variety of reservoir conditions.[1-10] Also, the advantages and disadvantages of these approaches are well documented in the literature.[11-15] Instead, I will focus on the computational efficiency of the two methods, especially for large-scale field applications. Of particular interest is the scaling of the computational costs for these two methods with increasing number of unknown parameters. Current industry practice involves generation of reservoir models consisting of several hundred thousand to millions of gridblocks. Integration of production data into such high resolution

---

This dissertation follows the style and format of the *SPE Journal.*

reservoir models can become computationally prohibitive. In this respect, the scaling behavior of the Bayesian vs. the deterministic approach can become a deciding factor in adopting one vs. the other approach.

The outline of this dissertation is as follows. First, I provide a brief mathematical background of the Bayesian and deterministic approaches as applied to production data integration into reservoir models. Second, I systematically investigate the scaling of the computational time for the two models with respect to the model size or the number of unknown parameters. Third, I propose an efficient and robust adaptation of the Bayesian formulation that can lead to orders of magnitude savings in computational time for model sizes larger than 100,000 gridblocks. My proposed method is based on an analytic computation of the square root of the inverse of the covariance matrix during production data integration using the Bayesian approach. I present a simple finite-difference stencil for the calculation of the square root of the inverse. This allows posing the Bayesian inverse problem in a manner analogous to the deterministic approach and the use of efficient sparse matrix solvers during the minimization of the data misfit. Fourth, I improve the accuracy and generality of the analytically-determined stencil by developing a numerically-determined stencil that preserves the computational convenience of the analytical one, but gets rid of its limitations. Finally, I propose an assessment of uncertainty technique that uses the power and utility of the proposed method along with an efficient technique to sample from the ensemble of conditioned realizations. The assessment of uncertainty technique is illustrated using a field example. The field application involves integration of water-cut response into the geologic model for the Goldsmith field in West Texas.

Background

In this section, I briefly review the Bayesian and the deterministic approaches to production data integration during reservoir characterization.

Bayesian Formulation. This approach follows from Bayes' rule and provides a natural framework for combining prior information related to the geologic model with the production data.[11] The goal is to derive a more refined statistical distribution for the model parameters, known as the posterior distribution which will be more tightly constrained compared to the prior distribution. We can then explore the posterior distribution to obtain plausible models given the data or simply use the posterior mean as the estimates and the posterior standard deviation as a measure of the 'confidence interval'.

During geostatistical reservoir modeling, it is common practice to assume that the spatial relationship between model parameters, for example gridblock permeabilities, is known via a prior covariance model derived from statistics of the field data. Furthermore, if we assume the prior model has a multivariate Gaussian distribution with a covariance matrix C_M and the production data has Gaussian uncertainty described by the data covariance C_d, then the Bayesian approach leads to the following posterior distribution[6]

$$P(m|d) \propto \exp\{-\frac{1}{2}[d-g(m)]^T C_d^{-1}[d-g(m)]+\frac{1}{2}[m-m_p]^T C_M^{-1}[m-m_p]\} \quad \text{.....(1.1)}$$

where d represents the data vector of dimension N, m represents the model parameter vector of dimension M and $g(m)$ defines the non-linear relationship between the model parameters and the calculated data.

Eq. 1.1 represents the model parameter distribution after we have assimilated the production data and prior information. The posterior distribution encompasses all possible solutions to the inverse problem; however, in practice we often derive the most probable or the maximum a posteriori estimate by minimizing the following misfit functional[2, 6, 7]

$$2F(m) = [d-g(m)]^T C_d^{-1}[d-g(m)]+[m-m_p]^T C_M^{-1}[m-m_p] \quad \text{.....(1.2)}$$

The minimization is typically carried out using a Gauss-Newton method that leads to the following iterative procedure[6, 7]

$$\delta m = (m_p - m) + C_M G^T \left[C_D + G C_M G^T\right]^{-1} \left[G(m - m_p) - (g(m) - d)\right] \quad \text{.....(1.3)}$$

In **Eq. 1.3**, G is the sensitivity matrix which contains the partial derivatives of the model response with respect to reservoir parameters.

The Bayesian approach is particularly well-suited for post-data inference as it assigns probability to the model space. The limitations are its reliance on the specification of the prior model statistics and also the computational cost in exploring the posterior distribution for high-dimensional problems.[13]

Deterministic Formulation. This approach (also referred to as the 'frequentist' method) utilizes only deterministic prior information about the model. However, the inference problem is still statistical as we will take into account random data uncertainties. Tikhonov's regularization, which imposes constraints on the model 'norm' and 'roughness,' is commonly used to restrict the family of models that fit the data.[16]

We start with a geologic model that already incorporates static reservoir information such as well logs and seismic data. We then minimize a penalized misfit function consisting of three terms as follows[9, 10]

$$\|\delta d - G\delta m\| + \beta_1 \|\delta m\| + \beta_2 \|\mathbf{L}\delta m\| \quad \text{.....(1.4)}$$

In the above expression, δd is the data residual vector, that is, the difference between the observed and the calculated production response, G is the sensitivity matrix that accounts for the change in production response because of a small change in reservoir properties such as permeability or porosity. Also, δm corresponds to the change in the

reservoir property and L is a second spatial difference operator. The first term is called the 'data misfit' term that minimizes the difference between the observed and calculated production response. The second term, 'norm constraint', ensures that the final model is not far from the initial model. This is justified because our prior model already contains sufficient geologic and static information related to the reservoir. Finally, the third term, 'roughness penalty' prefers smooth models that attempt to resolve large-scales structures rather than small-scale variations.

The minimum in **Eq. 1.4** can be obtained by an iterative least-squares solution to the augmented linear system [9, 10]

$$\begin{pmatrix} G \\ \beta_1 \mathbf{I} \\ \beta_2 \mathbf{L} \end{pmatrix} \delta m = \begin{pmatrix} \delta d \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \quad \text{.....(1.5)}$$

The weights β_1 and β_2 determine the relative strengths of the prior model and the roughness term. The selection of these weights can be somewhat subjective although there are guidelines in the literature.[10, 14] In general, the inversion results will be sensitive to these weights. We use an iterative sparse matrix solver, LSQR, for solving this augmented linear system efficiently.[17] The LSQR algorithm is well suited for highly ill-conditioned systems and has been widely used for large-scale tomographic problems in seismology.[18]

Thus, the idea behind the deterministic approach is to generate a set of models that satisfy the data and the prior deterministic criteria, for example, model norm and roughness constraints. Unlike the Bayesian approach, it does not rely on the knowledge of the prior model statistics; however, the relative weighting of the terms in **Eq. 1.4** can be subjective. The deterministic approach does not attach probability to the models and thus post-data inference can be difficult. For example, we cannot say that one particular model is more probable than the other.

Summary

Current techniques for production data integration into reservoir models can be broadly grouped into two categories: Bayesian and deterministic.

The Bayesian approach treats prior data as probabilistic; the deterministic approach treats it as fixed.

The advantage of the deterministic over the Bayesian approach is that it is computationally more efficient when optimized using a sparse matrix solver like the LSQR algorithm. This will be illustrated in Chapter II.

On the other hand, the Bayesian formulation has the advantage of accounting for uncertainty in both the production data and the prior model. This makes it the method of choice for the assessment of uncertainty.

The main objective of this dissertation then is to develop a technique that has the good traits of both techniques: the computational advantage of the deterministic formulation, and the capability of the Bayesian approach to treat the observed production data and prior reservoir information probabilistically.

Once this technique is developed, it will serve as the workhorse of a proposed methodology for the assessment of uncertainty.

CHAPTER II

COMPUTATIONAL SCALING PROPERTIES: BAYESIAN VS. DETERMINISTIC APPROACH

In this chapter, I compare the Bayesian and the deterministic formulations in terms of computational efficiency. As mentioned before, the current Bayesian formulation involves an iterative minimization via the Gauss-Newton algorithm. The deterministic formulation, on the other hand, uses **Eq. 1.5** together with the LSQR algorithm. I studied the scaling of the computation time for both approaches using a synthetic example that involves integration of water-cut response. The well configuration consists of a repeated 9-spot pattern as shown in **Fig. 2.1**. The objective is to integrate water-cut response from the 16 producing wells to the prior geological model using both approaches (i) deterministic approach (ii) Bayesian approach.

A series of inversions were carried out for the integration of water-cut data using both the Bayesian and the deterministic approach. A 3-D multiphase streamline simulator

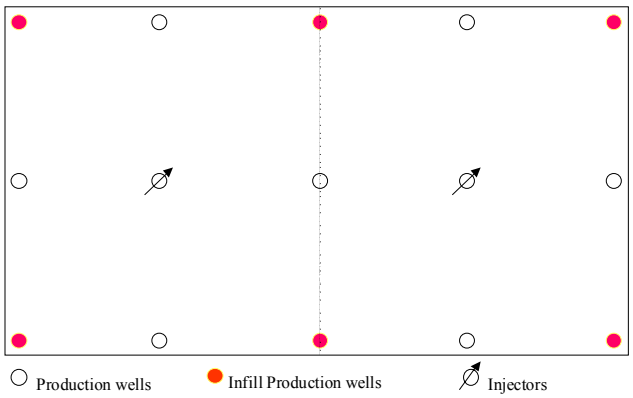


Fig. 2.1 – Well configuration for synthetic example.

was used to model the waterflood. The sensitivity of the water-cut response to gridblock permeabilities was obtained analytically using a single simulation.[9] **Table 1** summarizes

the cases examined with the number of gridblocks ranging from 1,250 to 125,000. These runs were carried out in an SGI ORIGIN 2000. We could not run larger models as we were limited by the memory requirements for the Bayesian approach.

Table 2.1 – Synthetic data sets. Number of gridblocks.

Set	N_x	N_y	N_z	M
1	50	25	1	1,250
2	50	25	10	12,500
3	50	25	20	25,000
4	50	25	40	50,000
5	50	25	100	125,000

Fig. 2.2 summarizes the CPU time per iteration required by the Bayesian and the deterministic formulations for the five cases. Clearly, the Bayesian not only requires more CPU time, but also this requirement grows at a faster rate than the deterministic approach. In particular, the deterministic approach shows a linear scaling behavior compared to a quadratic scaling for the Bayesian method. As a result, the Bayesian

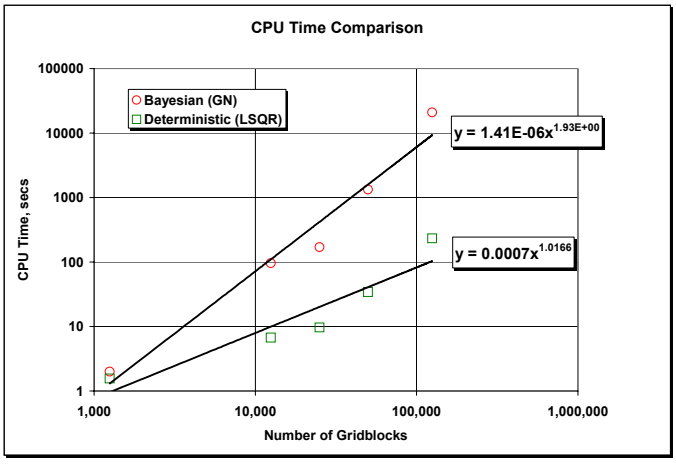


Fig. 2.2 - Comparison of CPU time between the Bayesian and the deterministic approaches.

approach requires orders of magnitude more computation time for models greater than 100,000 cells. For example, for Case 5, the Bayesian approach required 5 hours/iteration compared to 3 minutes/iteration for the deterministic approach; almost two orders of magnitude more computation time.

The scaling behavior of the methods can also be explained by examining the operation counts in these methods. For this purpose, I will assume a full sensitivity matrix and a full covariance matrix to estimate an upper bound on the operation count. The number of multiplications per iteration in the Bayesian approach using the Gauss-Newton algorithm can be calculated as follows:

- The product $GC_M G^T$ in **Eq. 1.3** must be computed in two steps:
  - GC_M **:** the total number of multiplications is NM^2 **.**
  - By the same token, multiplication of the matrix obtained above (GC_M) by the transpose of the sensitivity matrix, G^T, will require $N^2 M$ multiplications;
- The matrix $\left[C_d + GC_M G^T\right]$ is $N \times N$. Accordingly, N^3 multiplications are required to invert it;
- The product $C_M G^T$ will require NM^2 multiplications;
- The product of $C_M G^T$ by $\left[C_d + GC_M G^T\right]$ requires $N^2 M$ multiplications;
- The product of $G\left(m - m_p\right)$ requires NM multiplications;
- The product $C_M G^T \left[C_d + GC_M G^T\right]^{-1}$ times $G\left(m - m_p\right) - \left[g(m) - d\right]$ requires $N^2 M$ multiplications.

Adding up all of the above, results in the total number of multiplications per iteration required by the Bayesian approach using the Gauss-Newton algorithm

$$Z_{GN} = 2NM^2 + (3N^2 + N)M + N^3 \quad \text{.....(2.1)}$$

For the deterministic approach that uses the LSQR algorithm to solve the system of equations, the number of operations can be calculated as follows:

- Perform the multiplication Jv. It takes $M(N+c)$ iterations, where c is the number of cells in the stencil to account for the norm and the smoothness terms;
- Normalize column vector u. It takes $M+N$ multiplications;
- Perform the multiplication $J^T u$. It takes $M(N+c)$multiplications;
- Normalize column vector v. It requires M multiplications;
- Calculate column vectors x and w. It requires $2M$ multiplications.

Adding up all of the above, results in an expression for the total number of multiplications required by the deterministic approach using the LSQR algorithm

$$Z_{LSQR} = w \times [2M(N+c+2)+N] \quad \text{.....(2.2)}$$

From Eqs. 2.1 and 2.2, we can clearly see the quadratic vs. linear scaling with respect to the number of parameters for the Bayesian and the deterministic approaches, respectively.

Summary

It was shown both analytically and using synthetic data that the Bayesian approach using Gauss-Newton and the deterministic approach using the LSQR algorithm exhibit different computational scaling behaviors.

Thus, the CPU time required for data integration by the Bayesian method follows a quadratic trend when plotted as a function of the number of gridblocks. By contrast, the CPU time required by the deterministic method follows a linear trend.

Given the computational advantage of the deterministic approach, and the capability of the Bayesian formulation to describe the statistical properties of the data and the prior reservoir model, Chapter III explores the possibility of merging the advantages of both methods into a computationally efficient Bayesian formulation.

CHAPTER III

REFORMULATION OF THE SOLUTION TO THE BAYESIAN APPROACH

One of the objectives of this dissertation is to reformulate the solution of the Bayesian approach in a manner analogous to the deterministic formulation. This will allow us to preserve the statistical foundation of the Bayesian approach and also exploit the linear scaling of the deterministic approach which makes it particularly well-suited for large-scale inverse problems.

Optimization of the objective function of the Bayesian approach calls for determination of the maximum in the posterior distribution of the reservoir parameters, **Eq. 1.1.** Determination of this maximum is equivalent to the determination of the minimum of **Eq. 1.2.** This minimization is defined as the determination of the column vector of the reservoir parameters, m that satisfies

$$\nabla F(m) = 0 \quad \text{.....} \quad (3.1)$$

where, $\nabla F(m)$ is the gradient of $F(m)$ in the M-dimensional space. In other words, it is the vector of the partial derivatives of $F(m)$ with respect to the reservoir parameters in each gridblock. Consequently, minimization of $F(m)$ can be thought of as a root-finding exercise on **Eq. 3.1**.

Consider a solution vector, m, which is not a root of the function $\nabla F(m)$, but is "reasonably close" to a root. A Taylor series expansion of $\nabla F(m+\delta m)$ about m yields

$$\nabla F(m+\delta m) = \nabla F(m) + \nabla[\nabla F(m)]\delta m + \ldots \quad \text{.....} \quad (3.2)$$

If $\nabla F(m+\delta m)$ is set equal to zero, then $m+\delta m$ must be a root and the right-hand side of **Eq. 3.2** constitutes an equation for the root $m+\delta m$. Unfortunately, the equation is a

polynomial of degree infinity. However, an approximate value of the root $m+\delta m$ can be obtained by setting $\nabla F(m+\delta m)$ to zero and taking only the first two terms of the right-hand side of **Eq. 3.2** to yield

$$0 = \nabla F(m) + \nabla[\nabla F(m)]\delta m \quad \text{.....(3.3)}$$

To simplify the notation of **Eq. 3.3**, the following conventions are adopted:

$$q = \nabla F(m) \quad \text{.....(3.4)}$$

$$H = \nabla q = \nabla[\nabla F(m)] \quad \text{.....(3.5)}$$

where q and H are the gradient and the Hessian of $F(m)$, respectively.

Substituting **Eq. 3.4** and **Eq. 3.5** into **Eq. 3.3** gives

$$q + H\delta m = 0 \quad \text{.....(3.6)}$$

which can also be expressed as,

$$H\delta m = -q \quad \text{.....(3.7)}$$

Premultiplying both sides of **Eq. 3.7** by H^{-1} yields

$$\delta m = -H^{-1}q \quad \text{.....(3.8)}$$

Manipulation of **Eq. 1.2** and **Eq 3.8** can be simplified with the definition of two additional terms:

a. Residual. It is given the symbol e, and is defined as

$$e = \begin{bmatrix} C_d^{-1/2}\left(d - g(m)\right) \\ C_M^{-1/2}\left(m - m_p\right) \end{bmatrix} \quad \text{.....(3.9)}$$

b. Jacobian. It is given the symbol J , an is defined as the gradient of e

$$J = \nabla e \text{ .....(3.10)}$$

As a result, the objective function of the Bayesian formulation can be expressed as

$$F(m) = \frac{1}{2} e^T e \text{ .....(3.11)}$$

Taking the gradient of **Eq. 3.11**

$$q = J^T e \text{ .....(3.12)}$$

But the Hessian is the gradient of the gradient. Therefore, taking the gradient of **Eq. 3.12** leads to

$$\begin{aligned} H &= \nabla q = \nabla\left(J^T e\right) = \left(\nabla J^T\right)e + \left(\nabla e^T\right)J \\ &= Te + J^T J \end{aligned} \quad \text{.....(3.13)}$$

T is the Hessian of the residual. For small residuals (as we approximate the final answer), and quasilinear problems, the first term becomes negligible. Therefore, **Eq. 3.13** can be approximated to

$$H \cong J^T J \text{ .....(3.14)}$$

Substituting **Eq. 3.12** and **Eq. 3.14** into **Eq. 3.7**,

$$J^T J\delta m = -J^T e \quad \text{.....(3.15)}$$

Premultiplication of both sides of **Eq. 3.15** by $\left(J^T\right)^{-1}$ yields

$$J\delta m = -e \quad \text{.....(3.16)}$$

Eq. 3.9 can also be written as

$$e = \begin{bmatrix} C_d^{-1/2}\left(d - g(m)\right) \\ C_M^{-1/2}\left(m_p - m\right) \end{bmatrix} \quad \text{.....(3.17)}$$

Use of **Eq. 3.10** on **Eq. 3.17** yields the basic form of the Jacobian,

$$J = -\begin{bmatrix} C_d^{-1/2} G \\ C_M^{-1/2} \end{bmatrix} \quad \text{.....(3.18)}$$

where G is the sensitivity matrix and is equal to the gradient of $g(m)$.

Substituting **Eq. 3.17** and **Eq. 3.18** into **Eq. 3.16** gives the final expression for the iterative procedure of minimization of **Eq. 1.2** in vectorial form

$$\begin{bmatrix} C_d^{-1/2} G \\ C_M^{-1/2} \end{bmatrix} \delta m = \begin{bmatrix} C_d^{-1/2}\left(d - g(m)\right) \\ C_M^{-1/2}\left(m_p - m\right) \end{bmatrix} \quad \text{.....(3.19)}$$

Eq. 3.19 represents a system of equations that is analogous to the deterministic formulation in **Eq. 1.5,** except for the prior term on the right hand side. Thus, in principle, we can iteratively solve for the changes in the model parameter given by **Eq. 3.19** using the LSQR algorithm. However, difficulties arise in the computation of the square root of the matrix inverse in **Eq. 3.19**. In practice, the data covariance matrix is assumed to be diagonal and is thus easy to manipulate. However, the covariance matrix for the model parameters can be full and in general, the calculation of $C_M^{-1/2}$ will be computationally prohibitive for large scale inverse problems as shown in the next chapter. Hence, I will resort to an analytic approach for the computation of $C_M^{-1/2}$. The analytic approach, although completely general, is particularly well-suited for exponential models. This is one of the commonly used models in geostatistical reservoir characterization and leads to simple finite-difference stencils that allows for extremely efficient computation of the square root of the inverse of the covariance matrix.

Summary

The solution of the Bayesian approach was reformulated in a manner analogous to the deterministic formulation.

As a result, we have developed a powerful data integration formulation the can describe the statistical nature of the production data and the prior reservoir model, and analogously to the deterministic formulation, may display the linear trend of the CPU time required for the integration as a function of the number of gridblocks.

Given that this efficient formulation is a function of $C_M^{-1/2}$, which can be prohibitively expensive to calculate for large-scale applications, Chapter IV will present the development of an analytical stencil to approximate it in a reduced amount of time.

CHAPTER IV

NUMERICAL DETERMINATION OF THE SQUARE ROOT OF THE INVERSE

In the previous chapter, I presented a reformulation of the solution to the Bayesian approach analogous to that of the deterministic formulation. Its main advantage is that it is possible to preserve the statistical foundation of the Bayesian formulation while exploiting the linear scaling of the deterministic approach.

A salient feature of this reformulation is that it requires the calculation of the square root of the inverse of the covariance matrix of the reservoir parameters, $C_M^{-1/2}$. This chapter describes a technique to calculate $C_M^{-1/2}$ numerically.

As will be shown in a later chapter, as the number of gridblock approaches that of a field-scale problem, calculation of $C_M^{-1/2}$ using numerical techniques can become overwhelmingly prohibitive. Consequently, the next chapter will introduce an analytical technique to calculate $C_M^{-1/2}$ which will drastically reduce computation time. The numerical technique is presented here for comparison purposes only.

Sylvester's theorem provides a rapid way to calculate functions of a matrix. Some simple functions of a matrix of frequent occurrence are A^{-1} and A^N (for N large). However, for the purposes of this dissertation, we will only be interested in $N = -1/2$.

Before going into the somewhat abstract proof of Sylvester's theorem, I will take up a numerical example. Consider the matrix

$$A = \begin{bmatrix} 3 & -2 \\ 1 & 0 \end{bmatrix} \quad \text{.....(4.1)}$$

It will be necessary to have the column eigenvectors and the eigenvalues of this matrix; they are given by

$$\begin{bmatrix} 3 & -2 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = 1 \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \text{.....(4.2)}$$

$$\begin{bmatrix} 3 & -2 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = 2 \begin{bmatrix} 2 \\ 1 \end{bmatrix} \quad \text{.....(4.3)}$$

Since the matrix A is not symmetric, it has row eigenvectors which differ from the column vectors. These are

$$\begin{bmatrix} -1 & 2 \end{bmatrix} \begin{bmatrix} 3 & -2 \\ 1 & 0 \end{bmatrix} = 1 \begin{bmatrix} -1 & 2 \end{bmatrix} \quad \text{.....(4.4)}$$

$$\begin{bmatrix} 1 & -1 \end{bmatrix} \begin{bmatrix} 3 & -2 \\ 1 & 0 \end{bmatrix} = 2 \begin{bmatrix} 1 & -1 \end{bmatrix} \quad \text{.....(4.5)}$$

We may abbreviate **Eq. 4.2** through **Eq. 4.5** by

$$\begin{aligned} A\ c_1 &= \lambda_1\ c_1 \\ A\ c_2 &= \lambda_2\ c_2 \\ r_1\ A &= \lambda_1\ r_1 \\ r_2\ A &= \lambda_2\ r_2 \end{aligned} \quad \text{.....(4.6)}$$

It can be observed that r or c could be multiplied by an arbitrary scale factor and **Eq. 4.6** would still be valid. The eigenvectors are said to be normalized if scale factors have been chosen so that $r_1 \cdot c_1 = 1$ and $r_2 \cdot c_2 = 1$ .It will be observed that $r_1 \cdot c_2 = 0$ and $r_2 \cdot c_1 = 0$

Let us consider the behavior of the matrix $c_1 \cdot r_1$.

$$c_1 \cdot r_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \begin{bmatrix} -1 & 2 \end{bmatrix} = \begin{bmatrix} -1 & 2 \\ -1 & 2 \end{bmatrix} \quad \text{.....(4.7)}$$

Any power of this matrix is the matrix itself, for example, its square.

$$\begin{bmatrix} -1 & 2 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} -1 & 2 \\ -1 & 2 \end{bmatrix} = \begin{bmatrix} -1 & 2 \\ -1 & 2 \end{bmatrix} \quad \text{.....(4.8)}$$

This property is called idempotence (Latin for self-power). It arises because $(c_1 \cdot r_1)(c_1 \cdot r_1) = c_1 (r_1 \cdot c_1) r_1 = c_1 r_1$. The same thing, of course, is true for $c_2 \cdot r_2$. Now notice that the matrix $c_1 \cdot r_1$ is perpendicular to the matrix $c_2 \cdot r_2$, that is

$$\begin{bmatrix} 2 & -2 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} -1 & 2 \\ -1 & 2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad \text{.....(4.9)}$$

since r_1 and c_2 are perpendicular.

Sylvester's theorem says that any function f of the matrix A may be written

$$f(A) = f(\lambda_1) c_1 r_1 + f(\lambda_2) c_2 r_2 \quad \text{.....(4.10)}$$

The simplest example is

$$A = \lambda_1 c_1 r_1 + \lambda_2 c_2 r_2 \quad \text{.....(4.11)}$$

$$= 1 \begin{bmatrix} -1 & 2 \\ -1 & 2 \end{bmatrix} + 2 \begin{bmatrix} 2 & -2 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 3 & -2 \\ 1 & 0 \end{bmatrix} \quad \text{.....(4.12)}$$

Another example is

$$A^2 = \lambda_1^2 c_1 r_1 + \lambda_2^2 c_2 r_2 \quad \text{.....(4.13)}$$

$$A^2 = 1^2 \begin{bmatrix} -1 & 2 \\ -1 & 2 \end{bmatrix} + 2^2 \begin{bmatrix} 2 & -2 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 7 & -6 \\ 3 & -2 \end{bmatrix} \quad \text{.....(4.14)}$$

The inverse is

$$A^{-1} = 1^{-1} \begin{bmatrix} -1 & 2 \\ -1 & 2 \end{bmatrix} + 2^{-1} \begin{bmatrix} 2 & -2 \\ 1 & -1 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & 2 \\ -1 & 3 \end{bmatrix} \quad \text{.....(4.15)}$$

The square root of the inverse is

$$A^{-1/2} = 1^{-1/2} \begin{bmatrix} -1 & 2 \\ -1 & 2 \end{bmatrix} + 2^{-1/2} \begin{bmatrix} 2 & -2 \\ 1 & -1 \end{bmatrix} = \frac{\sqrt{2}}{2} \begin{bmatrix} 2-\sqrt{2} & -2+2\sqrt{2} \\ 1-\sqrt{2} & -1+2\sqrt{2} \end{bmatrix} \quad \text{.....(4.16)}$$

Let us see what it takes to prove Sylvester's theorem. We will need one basic result which is in all the books on matrix theory, namely, that most matrices can be diagonalized. In terms of our 2×2 example this takes the form

$$\begin{bmatrix} r_1 \\ r_2 \end{bmatrix} A \begin{bmatrix} c_1 \mid c_2 \end{bmatrix} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \quad \text{.....(4.17)}$$

where

$$\begin{bmatrix} r_1 \\ r_2 \end{bmatrix} \begin{bmatrix} c_1 \mid c_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \text{.....(4.18)}$$

Since a matrix commutes with its inverse, **Eq. 4.18** implies

$$\begin{bmatrix} c_1 \mid c_2 \end{bmatrix} \begin{bmatrix} r_1 \\ \hline r_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \text{.....(4.19)}$$

Premultiply **Eq. 4.17** by the column matrix and post multiply by the row matrix. Using **Eq. 4.19**, we get

$$A = \begin{bmatrix} c_1 \mid c_2 \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} r_1 \\ \hline r_2 \end{bmatrix} \quad \text{.....(4.20)}$$

Eq. 4.20 is **Eq. 4.11** in disguise, as we can see by writing **Eq. 4.20** as

$$\begin{aligned} A &= \begin{bmatrix} c_1 \mid c_2 \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} r_1 \\ \hline r_2 \end{bmatrix} + \begin{bmatrix} c_1 \mid c_2 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} r_1 \\ \hline r_2 \end{bmatrix} \quad \text{.....(4.21)} \\ &= \lambda_1 c_1 r_1 + \lambda_2 c_2 r_2 \end{aligned}$$

Now to get A^2 we have

$$A^2 = \left(\lambda_1 c_1 r_1 + \lambda_2 c_2 r_2\right)\left(\lambda_1 c_1 r_1 + \lambda_2 c_2 r_2\right) \quad \text{.....(4.22)}$$

Using the orthonormality of $c_1 r_1$ and $c_2 r_2$ this reduces to

$$A^2 = \lambda_1^2 c_1 r_1 + \lambda_2^2 c_2 r_2 \quad \text{.....(4.23)}$$

It is clear how **Eq. 4.20** can be used to prove Sylvester's theorem for any polynomial function of A. Clearly, there is nothing peculiar about 2×2 matrices either. This works for $n \times n$. Likewise, one may consider infinite series functions in A. Since almost any

function can be made up of infinite series, we can consider also transcendental functions like sine, cosine, exponential.

For the particular case of the covariance matrix, $A = C_M$ is a symmetrical matrix. Consequently, the matrix of row eigenvectors is equal to the transpose of the column eigenvectors. Thus, if V represents the matrix of the column eigenvectors

$$V = \begin{bmatrix} v_1 & v_2 & .... & v_M \end{bmatrix} \quad (4.24)$$

then, **Eq. 4.20** can be expressed as

$$A = V \Lambda V^T \quad (4.25)$$

where Λ represents the diagonal matrix of the eigenvalues of A

A general way of stating Sylvester's theorem is that when a matrix is raised to a power n, its eigenvectors remain unchanged, and the eigenvalues of the new matrix are equal to the eigenvalues of the original matrix raised to the n^{th} power.

$$A^n = V \Lambda^n V^T \quad (4.26)$$

For the particular case of $n = -1/2$, **Eq. 4.26** becomes

$$A^{-1/2} = V \Lambda^{-1/2} V^T \quad (4.27)$$

Equivalently,

$$A^{-1/2} = \lambda_1^{-1/2} v_1 v_1^T + \lambda_2^{-1/2} v_2 v_2^T + ... + \lambda_M^{-1/2} v_M v_M^T \quad (4.28)$$

Summary

The new Bayesian formulation presented in Chapter III has the potential to have the same computational scaling behavior as the deterministic formulation provided the $C_M^{-1/2}$ can also be determined in an efficient way. Numerical calculation of $C_M^{-1/2}$ can become prohibitive for large-scale problems, but it is presented in this chapter for three reasons:

1. As an option for small-scale problems,
2. To serve as a standard against which approximate techniques can be compared, and
3. To serve as the basis for the numerical-stencil technique presented in Chapter VII.

It is based on Sylvester's theorem which states the when a matrix is raised to a power n, its eigenvectors remain unchanged, and the eigenvalues of the new matrix are equal to the eigenvalues of the original matrix raised to the n^{th} power.

CHAPTER V

ANALYTIC DETERMINATION OF THE SQUARE ROOT OF THE INVERSE

Chapter IV presented a technique to numerically calculate the square root of a matrix inverse. Given the prohibitive computational cost of this calculation, use of the reformulation of the solution to the Bayesian approach presented in Chapter III would not be advantageous. This chapter presents an analytical technique to calculate the square root of the inverse of the covariance matrix for an exponential model. The development of this technique is intended to significantly reduce the computational cost.

Let $K = K(r)$ be the covariance kernel, and $\Gamma = \Gamma(r)$ the kernel of its inverse.

Also, let K be defined as[19]

$$K = \sigma^2 e^{-r} \quad \text{.....(5.1)}$$

where σ^2 is the variance of the reservoir parameters, and r is the normalized radial distance, defined as[19]

$$r = \sqrt{\left(\frac{x'}{a}\right)^2 + \left(\frac{y'}{b}\right)^2 + \left(\frac{z'}{c}\right)^2} \quad \text{.....(5.2)}$$

where $a, b,$ and c; and $x', y',$ and z' represent the ranges, and the coordinates in real space, respectively.

It is worth noting that it is common practice in geostatistics texts[20] to define the normalized radial distance as

$$r = \sqrt{\left(\frac{x'}{a'/3}\right)^2 + \left(\frac{y'}{b'/3}\right)^2 + \left(\frac{z'}{c'/3}\right)^2} \quad \text{.....(5.3)}$$

Again, for the sake of simplicity, the definition in **Eq. 5.2** will be used throughout the derivations. In the illustrative examples though, it will be useful to know what definition of range is being used. Thus,

$$\begin{aligned} a &= a'/3 \\ b &= b'/3 \\ c &= c'/3 \end{aligned} \quad \text{.....(5.4)}$$

Unlike the real space in which the ranges can be different in all three directions; in the normalized space, the range is always unity. In other words,

$$r = \sqrt{x^2 + y^2 + z^2} \quad \text{.....(5.5)}$$

where x, y, and z represent the Cartesian coordinate in the normalized space.

The notation with "primes" in the coordinates in real space was used so that the notation without "primes" could be reserved for the normalized coordinates. This will simplify the notation in the lengthy equations to come.

We can use convolution to describe the product of the matrices constructed using the kernels of the covariance and its inverse[19]

$$K * \Gamma = (\Delta x \Delta y \Delta z)\delta \quad \text{.....(5.6)}$$

where δ is the Dirac delta function; and $\Delta x, \Delta y, \Delta z$ are the grid sizes in the normalized space.

Taking the Fourier transform of **Eq. 5.6** yields,

$$\Im(K * \Gamma) = \Im(K)\Im(\Gamma) = \hat{K}\hat{\Gamma} = \Delta x \Delta y \Delta z \quad \text{.....(5.7)}$$

The importance of this result is that the convolution has been turned into a regular product in the Fourier domain. Consequently,

$$\hat{\Gamma} = \frac{\Delta x \Delta y \Delta z}{\hat{K}} \quad \text{.....(5.8)}$$

Taking the appropriate Hankel (3D Fourier) transform of **Eq. 5.1**,

$$\hat{K} = \frac{8\pi\sigma^2}{\left(1+\xi^2\right)^2} \quad \text{.....(5.9)}$$

Substituting **Eq. 5.9** into **Eq. 5.8**

$$\hat{\Gamma} = \frac{\Delta x \Delta y \Delta z}{8\pi\sigma^2}\left(1+\xi^2\right)^2 \quad \text{.....(5.10)}$$

Eq. 5.10 represents the kernel of the inverse of the covariance matrix of the reservoir parameters in the Fourier domain.

Since we are interested in the square root of the inverse of the covariance matrix, let us digress a little bit.

Like before, let Γ be the kernel of the inverse of the covariance matrix for an exponential model, and L be the kernel of its square root.

The product of $C_M^{-1/2}$ with itself can be described using the convolution of its kernel with itself as

$$L * L = \Gamma \quad \text{.....(5.11)}$$

Once again, taking the Fourier transform of **Eq. 5.11** turns the convolution into a regular product in the Fourier domain. Thus,

$$\hat{L}^2 = \hat{\Gamma} \quad \text{.....(5.12)}$$

Substituting **Eq.5.10** into **Eq. 5.12**,

$$\hat{L}^2 = \frac{\Delta x \Delta y \Delta z}{8\pi\sigma^2}\left(1+\xi^2\right)^2 \quad \text{.....(5.13)}$$

Taking the square root of **Eq. 5.13**

$$\hat{L} = \sqrt{\frac{\Delta x \Delta y \Delta z}{8\pi\sigma^2}}\left(1+\xi^2\right) \quad \text{.....(5.14)}$$

Before inversion of **Eq. 5.14** from the Fourier domain is attempted, some digression is necessary. For an arbitrary function $f = f(r)$, let

$$\nabla f = \frac{df}{dr} \quad \text{.....(5.15)}$$

Taking the Fourier transform of **Eq. 5.15**

$$\Im(\nabla f) = \int_{-\infty}^{\infty} \nabla f \, e^{-i\xi r} \, dr \quad \text{.....(5.16)}$$

Using the technique of integration by parts on **Eq. 5.16** leads to

$$\Im(\nabla f) = f \, e^{-i\xi r} \, |_{-\infty}^{\infty} + i\xi \int_{-\infty}^{\infty} f \, e^{-i\xi r} \, dr \quad \text{.....(5.17)}$$

Since r is always positive—see **Eq. 5.2**, and f is a bounded function, the first term in **Eq. 5.17** is zero, leading to

$$\Im(\nabla f) = i\xi\,\hat{f} \quad \text{.....(5.18)}$$

Recursive use of the technique of integration by parts leads to the following general rule:

$$\Im(\nabla^n f) = (i)^n (\xi)^n\ \hat{f} \quad \text{.....(5.19)}$$

Another important result, already used to transform **Eq. 5.6**, is that,

$$\Im(\delta) = \hat{\delta} = 1 \quad \text{.....(5.20)}$$

Using **Eq. 5.19** and **Eq. 5.20**, **Eq. 5.14** can be back transformed to become

$$L = \sqrt{\frac{\Delta x \Delta y \Delta z}{8\pi\sigma^2}}\left(\delta(r) - \nabla^2\delta(r)\right) \quad \text{.....(5.21)}$$

Eq. 5.21 is the kernel of the square root of the inverse of the covariance matrix of the reservoir parameters for an exponential model in an anisotropic 3-dimensional medium, where

$$\nabla^2\delta = \frac{d^2\delta}{dr^2} \quad \text{.....(5.22)}$$

Using the chain rule of differentiation

$$\nabla^2\delta = h^2\frac{\partial^2\delta}{\partial x^2} + 2hk\frac{\partial^2\delta}{\partial x\partial y} + k^2\frac{\partial^2\delta}{\partial y^2} + 2hl\frac{\partial^2\delta}{\partial x\partial z} + 2kl\frac{\partial^2\delta}{\partial y\partial z} + l^2\frac{\partial^2\delta}{\partial z^2} \quad \text{.....(5.23)}$$

h, k and l are the components of a unit vector in the vector field that points away from the center of every normalized gridblock. Since I will be using finite differences to approximate the partial derivatives in **Eq. 5.23**, the vector field will also be an approximation using only the vectors pointing from the given gridblock to each component of the resulting stencil.

Mathematically, the components of this unit vector are defined as:

$$h = \frac{dx}{dr} \quad \text{.....(5.24a)}$$

$$k = \frac{dy}{dr} \quad \text{.....(5.24b)}$$

$$l = \frac{dz}{dr} \quad \text{.....(5.24c)}$$

Eq. 5.25 shows the central difference approximations of the partial derivatives in **Eq. 5.23**

$$h^2\frac{\partial^2\delta}{\partial x^2} = \frac{h^2}{(\Delta x)^2}\left(\delta_{i-1} - 2\delta_i + \delta_{i+1}\right) \quad \text{.....(5.25a)}$$

$$k^2\frac{\partial^2\delta}{\partial y^2} = \frac{k^2}{(\Delta y)^2}\left(\delta_{j-1} - 2\delta_j + \delta_{j+1}\right) \quad \text{.....(5.25b)}$$

$$l^2 \frac{\partial^2 \delta}{\partial z^2} = \frac{l^2}{(\Delta z)^2}\left(\delta_{k-1} - 2\delta_k + \delta_{k+1}\right) \quad \text{.....(5.25c)}$$

$$2hk \frac{\partial^2 \delta}{\partial x \partial y} = \frac{hk}{2\Delta x \Delta y}\left(\delta_{i-1,j-1} - \delta_{i+1,j-1} - \delta_{i-1,j+1} + \delta_{i+1,j+1}\right) \quad \text{.....(5.25d)}$$

$$2hl \frac{\partial^2 \delta}{\partial x \partial z} = \frac{hl}{2\Delta x \Delta z}\left(\delta_{i-1,k-1} - \delta_{i+1,k-1} - \delta_{i-1,k+1} + \delta_{i+1,k+1}\right) \quad \text{.....(5.25e)}$$

$$2kl \frac{\partial^2 \delta}{\partial y \partial z} = \frac{kl}{2\Delta y \Delta z}\left(\delta_{j-1,k-1} - \delta_{j+1,k-1} - \delta_{j-1,k+1} + \delta_{j+1,k+1}\right) \quad \text{.....(5.25f)}$$

To simplify the notation, let

$$\Omega = \sqrt{\frac{\Delta x \Delta y \Delta z}{8\pi\sigma^2}} \quad \text{.....(5.26)}$$

Substituting **Eq. 5.23** into **Eq. 5.21**

$$L = \Omega\left(\delta - h^2 \frac{\partial^2 \delta}{\partial x^2} - k^2 \frac{\partial^2 \delta}{\partial y^2} - l^2 \frac{\partial^2 \delta}{\partial z^2} - hk \frac{\partial^2 \delta}{\partial x \partial y} - hl \frac{\partial^2 \delta}{\partial x \partial z} - kl \frac{\partial^2 \delta}{\partial y \partial z}\right) \quad \text{.....(5.27)}$$

Two important concepts are relevant to simplify the calculation of the components of the unit vector (h, k and l in **Eq. 5.24**). These concepts can be summarized as two rules: sign rules and magnitude rules.

a. Sign Rules. As stated above, h, k and l are components of a unit vector, and their sign must be accounted for. Given that the unit vector with components h, k and l originates at the grid block identified by the sub indices i, j and k, two different cases must be considered:

- Even powers of h, k and l are always positive regardless of the sign of the component. Thus, $h^2 = |h|^2$, $k^2 = |k|^2$ and $l^2 = |l|^2$

- The combination of powers of h, k and l may be either positive or negative.
  - ✓ Consider that the unit vector points to the $i-1, j-1, k$ gridblock. Then, both h and k will be negative. As a result, $hk = |h||k|$

    The same will happen if it points to $i+1, j+1, k$: both h and k will be positive. Consequently, $hk = |h||k|$
  - ✓ However, if the unit vector points to $i-1, j+1, k$, h will be negative, and k will be positive. Therefore, $hk = -|h||k|$

b. Magnitude Rules. In the context of finite differences, **Eq. 5.24** can be approximated to $|h| = \frac{\Delta x}{\Delta r}$, $|k| = \frac{\Delta y}{\Delta r}$ and $|l| = \frac{\Delta z}{\Delta r}$. As a result, $\frac{|h|}{\Delta x} = \frac{1}{\Delta r}$, $\frac{|k|}{\Delta y} = \frac{1}{\Delta r}$ and $\frac{|l|}{\Delta z} = \frac{1}{\Delta r}$. However, the way in which Δr is calculated will depend on the context. For example, if we are calculating $\frac{|k||l|}{(\Delta y)(\Delta z)}$ in the context of $\delta_{i,j-1,k-1}$, Δr will be calculated as

$$\frac{|k||l|}{\Delta y(\Delta z)} = \frac{(\Delta y)(\Delta z)}{(\Delta y)(\Delta z)\left(\sqrt{(\Delta y)^2 + (\Delta z)^2}\right)\left(\sqrt{(\Delta y)^2 + (\Delta z)^2}\right)} = \frac{1}{(\Delta r)(\Delta r)}$$

As a result,

$$\Delta r = \sqrt{(\Delta y)^2 + (\Delta z)^2}$$

To identify radial distances, a subscript notation was used. Thus, for this example, Δr is interchangeable with r_{yz} .

The listing below defines all the radial distances required for construction of the stencil:

$$r_x = \Delta x \quad \text{.....(5.28a)}$$

$$r_y = \Delta y \quad \text{..... (5.28b)}$$

$$r_z = \Delta z \quad \text{.....(5.28c)}$$

$$r_{xy} = \sqrt{(\Delta x)^2 + (\Delta y)^2} \quad \text{..... (5.28d)}$$

$$r_{xz} = \sqrt{(\Delta x)^2 + (\Delta z)^2} \quad \text{.....(5.28e)}$$

$$r_{yz} = \sqrt{(\Delta y)^2 + (\Delta z)^2} \quad \text{.....(5.28f)}$$

Substituting **Eq. 5.25** into **Eq. 5.27** and collecting the terms that multiply the Dirac delta function in different gridblocks

$$\delta_{i,j-1,k-1} : -\Omega \frac{kl}{2(\Delta y)(\Delta z)} \quad \text{.....(5.29a)}$$

$$\delta_{i-1,j,k-1} : -\Omega \frac{hl}{2(\Delta x)(\Delta z)} \quad \text{..... (5.29b)}$$

$$\delta_{i,j,k-1} : -\Omega \frac{l^2}{(\Delta z)^2} \quad \text{.....(5.29c)}$$

$$\delta_{i+1,j,k-1} : \Omega \frac{hl}{2(\Delta x)(\Delta z)} \quad \text{..... (5.29d)}$$

$$\delta_{i,j+1,k-1} : \Omega \frac{kl}{2(\Delta y)(\Delta z)} \quad \text{.....(5.29e)}$$

$$\delta_{i-1,j-1,k} : -\Omega \frac{hk}{2(\Delta x)(\Delta y)} \quad \text{.....(5.29f)}$$

$$\delta_{i,j-1,k} : -\Omega \frac{k^2}{(\Delta y)^2} \quad \text{..... (5.29g)}$$

$$\delta_{i+1,j-1,k} : \Omega \frac{hk}{2(\Delta x)(\Delta y)} \quad \text{..... (5.29h)}$$

$$\delta_{i-1,j,k} : -\Omega \frac{h^2}{(\Delta x)^2} \quad \text{..... (5.29i)}$$

$$\delta_{i,j,k} : \Omega \left[1 + \frac{2h^2}{(\Delta x)^2} + \frac{2k^2}{(\Delta y)^2} + \frac{2l^2}{(\Delta z)^2} \right] \quad \text{..... (5.29j)}$$

$$\delta_{i+1,j,k} : -\Omega \frac{h^2}{(\Delta x)^2} \quad \text{..... (5.29k)}$$

$$\delta_{i-1,j+1,k} : \Omega \frac{hk}{2(\Delta x)(\Delta y)} \quad \text{..... (5.29 l)}$$

$$\delta_{i,j+1,k} : -\Omega \frac{k^2}{(\Delta y)^2} \quad \text{..... (5.29m)}$$

$$\delta_{i+1,j+1,k} : -\Omega \frac{hk}{2(\Delta x)(\Delta y)} \quad \text{..... (5.29n)}$$

$$\delta_{i,j-1,k+1} : \Omega \frac{kl}{2(\Delta y)(\Delta z)} \quad \text{..... (5.29o)}$$

$$\delta_{i-1,j,k+1} : \Omega \frac{hl}{2(\Delta x)(\Delta z)} \quad \text{..... (5.29p)}$$

$$\delta_{i,j,k+1} : -\Omega \frac{l^2}{(\Delta z)^2} \quad \text{..... (5.29q)}$$

$$\delta_{i+1,j,k+1} : -\Omega \frac{hl}{2(\Delta x)(\Delta z)} \quad \text{.....(5.29r)}$$

$$\delta_{i,j+1,k+1} : -\Omega \frac{kl}{2(\Delta y)(\Delta z)} \quad \text{.....(5.29s)}$$

Applying the sign and the magnitude rules to **Eq. 5.29**

$$\delta_{i,j-1,k-1} : -\Omega \frac{|k||l|}{2(\Delta y)(\Delta z)} = -\frac{\Omega}{2r_{yz}^2} \quad \text{.....(5.30a)}$$

$$\delta_{i-1,j,k-1} : -\Omega \frac{|h||l|}{2(\Delta x)(\Delta z)} = -\frac{\Omega}{2r_{xz}^2} \quad \text{..... (5.30b)}$$

$$\delta_{i,j,k-1} : -\Omega \frac{|l|^2}{(\Delta z)^2} = -\frac{\Omega}{r_z^2} \quad \text{.....(5.30c)}$$

$$\delta_{i+1,j,k-1} : -\Omega \frac{|h||l|}{2(\Delta x)(\Delta z)} = -\frac{\Omega}{2r_{xz}^2} \quad \text{..... (5.30d)}$$

$$\delta_{i,j+1,k-1} : -\Omega \frac{|k||l|}{2(\Delta y)(\Delta z)} = -\frac{\Omega}{2r_{yz}^2} \quad \text{.....(5.30e)}$$

$$\delta_{i-1,j-1,k} : -\Omega \frac{|h||k|}{2(\Delta x)(\Delta y)} = -\frac{\Omega}{2r_{xy}^2} \quad \text{..... (5.30f)}$$

$$\delta_{i,j-1,k} : -\Omega \frac{|k|^2}{(\Delta y)^2} = -\frac{\Omega}{r_y^2} \quad \text{..... (5.30g)}$$

$$\delta_{i+1,j-1,k} : -\Omega \frac{|h||k|}{2(\Delta x)(\Delta y)} = -\frac{\Omega}{2r_{xy}^2} \quad \text{..... (5.30h)}$$

$$\delta_{i-1,j,k} : -\Omega \frac{|h|^2}{(\Delta x)^2} = -\frac{\Omega}{r_x^2} \quad \text{..... (5.30i)}$$

$$\delta_{i,j,k} : \Omega\left[1 + \frac{2|h|^2}{(\Delta x)^2} + \frac{2|k|^2}{(\Delta y)^2} + \frac{2|l|^2}{(\Delta z)^2}\right] = \Omega\left(1 + \frac{2}{r_x^2} + \frac{2}{r_y^2} + \frac{2}{r_z^2}\right) \quad \text{..... (5.30j)}$$

$$\delta_{i+1,j,k} : -\Omega \frac{|h|^2}{(\Delta x)^2} = -\frac{\Omega}{r_x^2} \quad \text{..... (5.30k)}$$

$$\delta_{i-1,j+1,k} : -\Omega \frac{|h||k|}{2(\Delta x)(\Delta y)} = -\frac{\Omega}{2r_{xy}^2} \quad \text{..... (5.30 l)}$$

$$\delta_{i,j+1,k} : -\Omega \frac{|k|^2}{(\Delta y)^2} = -\frac{\Omega}{r_y^2} \quad \text{..... (5.30m)}$$

$$\delta_{i+1,j+1,k} : -\Omega \frac{|h||k|}{2(\Delta x)(\Delta y)} = -\frac{\Omega}{2r_{xy}^2} \quad \text{..... (5.30n)}$$

$$\delta_{i,j-1,k+1} : -\Omega \frac{|k||l|}{2(\Delta y)(\Delta z)} = -\frac{\Omega}{2r_{yz}^2} \quad \text{..... (5.30o)}$$

$$\delta_{i-1,j,k+1} : -\Omega \frac{-|h||l|}{2(\Delta x)(\Delta z)} = -\frac{\Omega}{2r_{xz}^2} \quad \text{..... (5.30p)}$$

$$\delta_{i,j,k+1} : -\Omega \frac{|l|^2}{(\Delta z)^2} = -\frac{\Omega}{r_z^2} \quad \text{..... (5.30q)}$$

$$\delta_{i+1,j,k+1} : -\Omega \frac{|h||l|}{2(\Delta x)(\Delta z)} = -\frac{\Omega}{2r_{xz}^2} \quad \text{..... (5.30r)}$$

$$\delta_{i,j+1,k+1} : -\Omega \frac{|k||l|}{2(\Delta y)(\Delta z)} = -\frac{\Omega}{2r_{yz}^2} \quad \text{..... (5.30s)}$$

The resulting stencil is composed of 19 cells. However, because of symmetry, only 7 different terms, C_1 to C_7, as listed in **Eq. 5.31**, are required. Their spatial distribution is illustrated in **Fig. 5.1**.

$$C_1 = -\frac{\Omega}{2r_{yz}^2} \quad \text{..... (5.31a)}$$

$$C_2 = -\frac{\Omega}{2r_{xz}^2} \quad \text{..... (5.31b)}$$

$$C_3 = -\frac{\Omega}{2r_{xy}^2} \quad \text{..... (5.31c)}$$

$$C_4 = -\frac{\Omega}{r_x^2} \quad \text{..... (5.31d)}$$

$$C_5 = -\frac{\Omega}{r_y^2} \quad \text{..... (5.31e)}$$

$$C_6 = -\frac{\Omega}{r_z^2} \quad \text{..... (5.31f)}$$

$$C_7 = \Omega\left(1+\frac{2}{r_x^2}+\frac{2}{r_y^2}+\frac{2}{r_z^2}\right) \quad \text{..... (5.31g)}$$

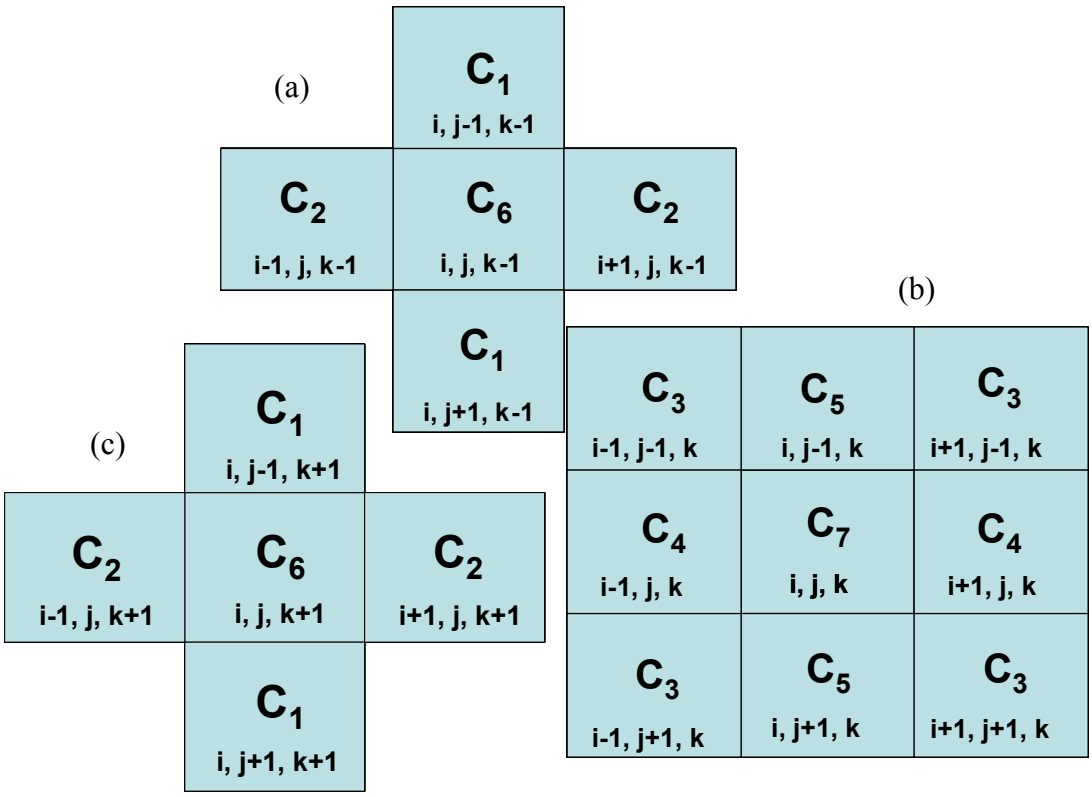


Fig. 5.1 – Generalized 3-dimensional stencil. (a) Layer $k-1$, (b) Layer k, (c) Layer $k+1$.

A further simplification can be obtained by noting that in the normalized space, we are dealing with a spherical covariance model of range equal to unity, rather than an ellipsoid of ranges a, b and c. Consequently,

$$\frac{\Delta x'}{a} = \frac{\Delta y'}{b} = \frac{\Delta z'}{c} = \Delta \quad \text{.....(5.32a)}$$

Or equivalently,

$$\Delta x = \Delta y = \Delta z = \Delta \quad \text{..... (5.32b)}$$

The importance of this observation is that it allows **Eq. 5.31** to be simplified to only 3 terms:

$$C_1 = C_2 = C_3 = -\frac{\Omega}{4\Delta^2} = S_1 \quad \text{.....(5.33a)}$$

$$C_4 = C_5 = C_6 = -\frac{\Omega}{\Delta^2} = S_2 \quad \text{..... (5.33b)}$$

$$C_7 = \Omega\left(1 + \frac{6}{\Delta^2}\right) = S_3 \quad \text{.....(5.33c)}$$

Making use of the observation in **Eq. 5.32**, Ω can be simplified to

$$\Omega = \sqrt{\frac{\Delta^3}{8\pi\sigma^2}} \quad \text{..... (5.34)}$$

Substitution of **Eq. 5.34** into **Eq. 5.33** leads to the final form of the 3-dimensional anisotropic stencil shown in **Eq. 5.35**, and graphically illustrated in **Fig. 5.2**.

$$S_1 = -\frac{\Psi}{4\Delta^{1/2}} \quad \text{.....(5.35a)}$$

$$S_2 = -\frac{\Psi}{\Delta^{1/2}} \quad \text{..... (5.35b)}$$

$$S_3 = \Psi\left(\Delta^{3/2} + \frac{6}{\Delta^{1/2}}\right) \quad \text{.....(5.35c)}$$

where

$$\Psi = \frac{1}{\sqrt{8\pi\sigma^2}} \quad \text{..... (5.36)}$$

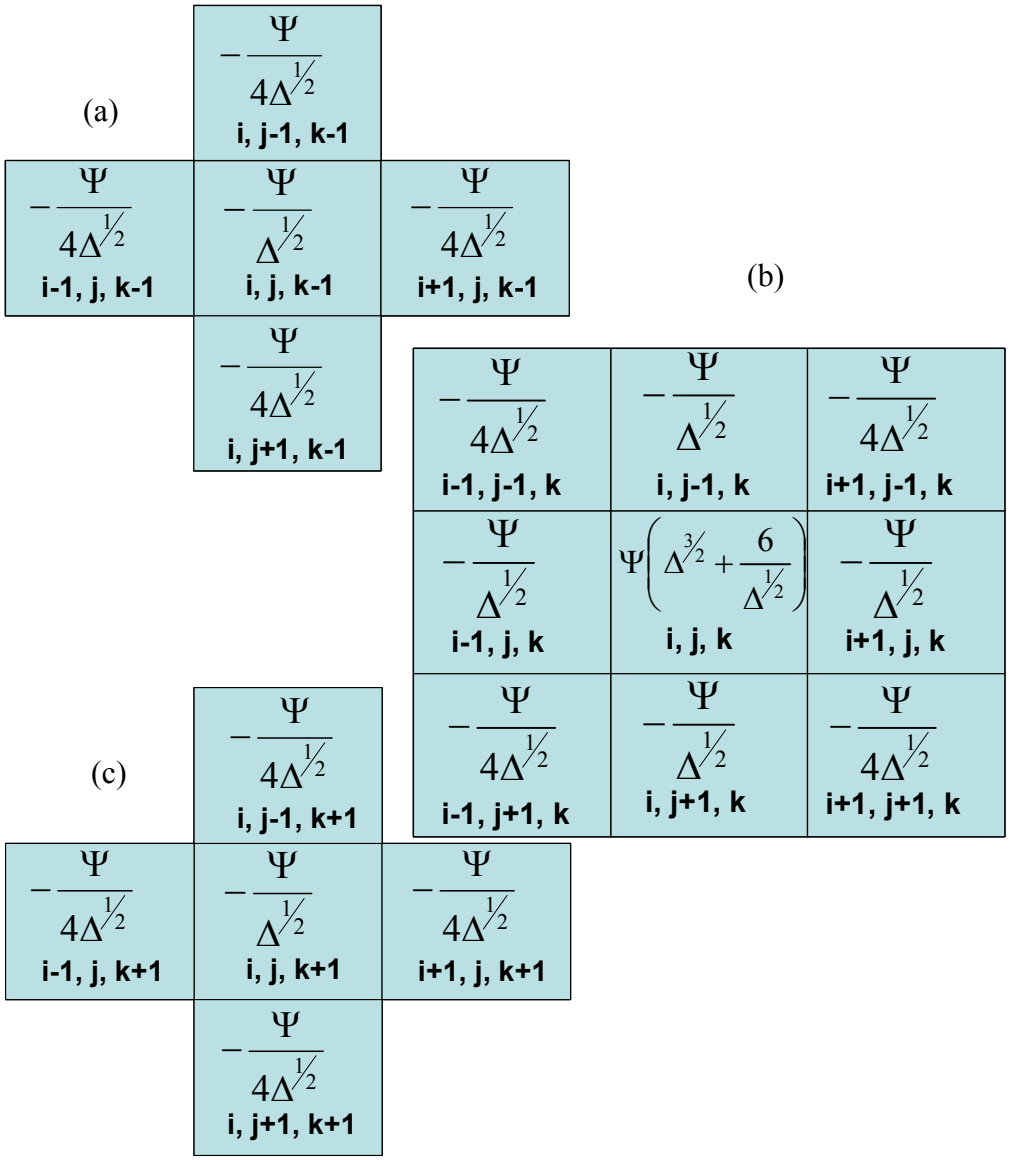


Fig. 5.2 – 3-dimensional stencil. (a) Layer $k-1$, (b) Layer k, (c) Layer $k+1$.

It is important to note that the observation stated in **Eq. 5.32** is a result of the normalization of the spatial coordinates. Although it might appear to be a limitation to the technique, this point can be examined from two different perspectives:

a. From an intuitive point of view, it is obvious that the size of the grid should be proportional to the range displayed by the data in a given direction.
b. An entirely different practice consists of relying on one's personal judgment to allocate grid sizes in every direction regardless of what is dictated by the spatial correlation of the data.

The first perspective allows the use of the stencil technique presented here. The second one would require the use of the numerical technique presented in Chapter IV. The comparison of CPU times between both techniques (Chapter VI) will show that the analytical stencil can be several orders of magnitude faster than the numerical method. Chapter VII will present a generalized technique to approximate the square root of the inverse of the covariance matrix that incorporates both ideas: description of the kernel using a numerically determined stencil.

The following thought experiment shows how the constraint in **Eq. 5.32** can be combined with three of the four constraints inherent to every inversion problem.

When tackling an inversion problem, there are only four actual constraints: the size of the system (reservoir size), the maximum allowed number of grid blocks, the extent of the anisotropy, and the spread of the reservoir parameters. Let

L_x : Length in the x direction,

L_y : Length in the y direction,

L_z : Length in the z direction,

M : Maximum allowed number of grid blocks, which can be expressed as

N_x : Number of grid blocks in the x direction,

N_y : Number of grid blocks in the y direction, and

N_z : Number of grid blocks in the z direction.

In other words,

$$M = N_x N_y N_z \quad \text{..... (5.37)}$$

$$N_x = \frac{L_x}{\Delta x'} \quad \text{.....(5.38a)}$$

$$N_y = \frac{L_y}{\Delta y'} \quad \text{..... (5.38b)}$$

$$N_z = \frac{L_z}{\Delta z'} \quad \text{.....(5.38c)}$$

Substituting **Eq. 5.38** into **Eq. 5.37**

$$M = \frac{L_x L_y L_z}{\Delta x' \Delta y' \Delta z'} \quad \text{..... (5.39)}$$

From **Eq. 5.32a**, the grid block sizes in real space are defined as

$$\Delta x' = a\Delta \quad \text{.....(5.40a)}$$

$$\Delta y' = b\Delta \quad \text{..... (5.40b)}$$

$$\Delta z' = c\Delta \quad \text{.....(5.40c)}$$

Substituting **Eq. 5.40** into **Eq. 5.39**,

$$M = \frac{L_x L_y L_z}{abc\Delta^3}$$

Solving for Δ,

$$\Delta = \sqrt[3]{\frac{L_x L_y L_z}{abcM}} \quad \text{..... (5.41)}$$

In summary, Δ reduces the three geometrical (distance-related) constraints (size of the system, maximum number of gridblocks, and extent of anisotropy) into a single geometrical constraint.

As a result, design of the grid (**Eq. 5.40**) and construction of the corresponding inverse of the covariance matrix (**Eq. 5.35**, **Eq. 5.36** and **Fig. 5.2**) can be performed simultaneously as a function of their only two constraints: the geometrical constraint, Δ, and the statistical constraint, σ^2.

Illustrative Example

This example will illustrate two points:

- Design of the grid,
- Check the validity of the approximate calculation of the square root of the inverse.

Say we have a reservoir with dimensions

$L_x = 50\ ft$,

$L_y = 70\ ft$, and

$L_z = 3\ ft$.

The ranges in the x', y' and z' directions are

$a = 100\ ft$,

$b = 90\ ft$,

$c = 10\ ft$.

The variance of the data is 14 units squared.

I will assume that I can afford to use more than 34, but no more than 50 gridblocks.

Then, using **Eq. 5.41**, with $M = 34$,

$\Delta = 0.1508$

And using **Eq. 5.40**,

$\Delta x' = 15.1\, ft$

$\Delta y' = 13.6\, ft$

$\Delta z' = 1.5\, ft$

However, upon calculation of the number of gridblocks in each direction, an expected inconsistency is observed:

$N_x = 3.3150$

$N_y = 5.1566$

$N_z = 1.9890$

Namely, the number of gridblocks is not an integer number, which is not physically realistic. Fortunately, this is a phenomenon that has been tackled by the simulation community over the years, and simple tricks are available to get around it. All that needs to be done is to stretch the dimensions of the reservoir by the right amount to make N_x, N_y and N_z integer numbers. Since this will affect the pore volume and the transmissibility of the gridblocks on the outer boundary cells of the system (reservoir), relevant corrections are made in the forward model.

The corrected number of gridblocks can be calculated from

$N_{xc} = \text{int}(N_x + 1)$ .....(5.42a)

$N_{yc} = \text{int}(N_y + 1)$ ..... (5.42b)

$N_{zc} = \text{int}(N_z + 1)$ .....(5.42c)

Correspondingly, the corrected reservoir dimensions can be determined from:

$$L_{xc} = \left(\frac{N_{xc}}{N_x} \right) L_x \quad \text{.....(5.43a)}$$

$$L_{yc} = \left(\frac{N_{yc}}{N_y} \right) L_y \quad \text{..... (5.43b)}$$

$$L_{zc} = \left(\frac{N_{zc}}{N_z} \right) L_z \quad \text{.....(5.43c)}$$

Thus, for the current example,

$N_{xc} = 4$,

$N_{yc} = 6$,

$N_{zc} = 2$, and

$M_c = 48$

The gridblock sizes remain unchanged, and the corrected dimensions of the reservoir become:

$L_{xc} = 60.33\ ft$,

$L_{yc} = 81.45\ ft$, and

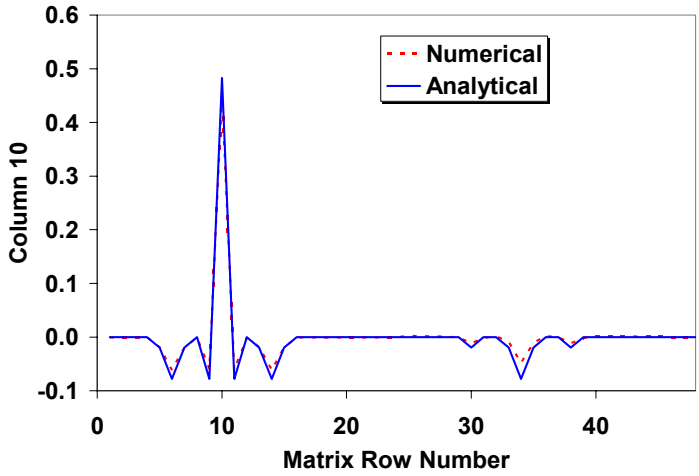
$L_{xc} = 3.02\ ft$.

Now, the data are set to calculate the square root of the inverse of the covariance matrix using the stencil presented in **Eq. 5.35**, **Eq. 5.36** and **Fig.5.2**. **Fig. 5.3** shows the comparison between both techniques: the exact square root of the inverse of the

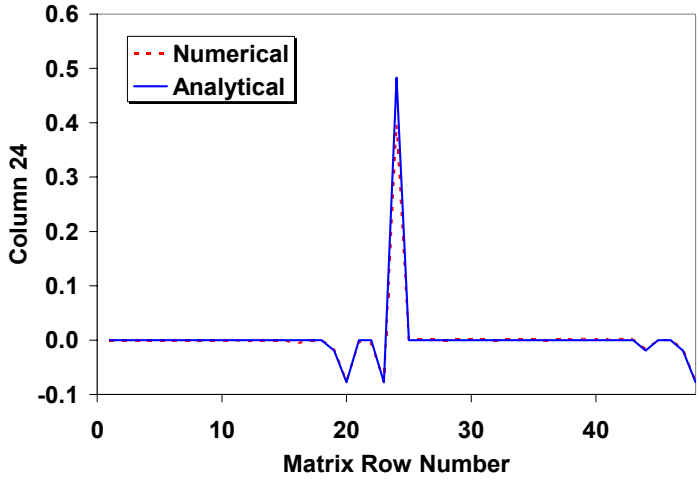
covariance matrix—calculated using the technique outlined in Chapter IV—and the approximation obtained using the 3D anisotropic stencil presented in this chapter.

Fig. 5.3 illustrates that the 3D anisotropic stencil technique turned out to be an excellent tool to approximate the square root of the inverse of the covariance matrix when the covariance of the reservoir parameters can be modeled using an exponential model.

Chapter VI will show the astronomical computational advantage of the new technique in terms of CPU memory and time.

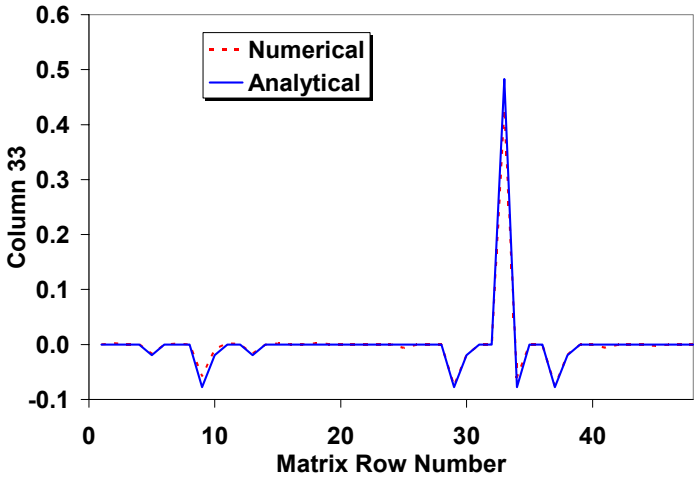


(a)



(b)

Fig. 5.3 – Numerically- and analytically-determined square root of the inverse matrix compare remarkably well. (a) column 10, (b) column 24, (c) column 33.



(c)

Fig. 5.3 – (continued)

Summary

Chapter IV presented a numerical technique to calculate $C_M^{-1/2}$. This matrix is a major component in the new Bayesian formulation presented in Chapter III. Given that the numerical technique becomes computationally prohibitive for large-scale problems, this chapter presented a technique to approximate $C_M^{-1/2}$ using an analytically determined stencil.

The computational advantage of this analytical stencil over the numerical technique is illustrated in Chapter VI.

CHAPTER VI

COMPUTATIONAL ADVANTAGE OF THE ANALYTICAL 3D STENCIL OVER NUMERICAL TECHNIQUE

From the realization that the product of two matrices can be described using the mathematical concept of convolution, it was possible to develop an analytical stencil that allows the determination of the square root of the inverse of the covariance matrix for an exponential model. Using an example, the "goodness" of the approximation was illustrated.

The purpose of this chapter is to compare their scaling properties. I made this comparison from two different perspectives. The first one compares the time required to generate the square root of the inverse only. The second one compares the whole inversion process.

Increased Efficiency in the Calculation of the Square Root of the Inverse

Continuing with the same example problem presented in Chapter V, **Table 6.1** is a reminder of three of the four constraints of the inversion problem: anisotropy, reservoir size, and variance of the parameters. The number of grid blocks will be the variable in this chapter.

Table 6.1 – Reservoir size and anisotropy, and parameter spread.

L_x, ft	50
L_y, ft	70
L_z, ft	3
a, ft	100
b, ft	90
c, ft	10
σ^2, unit^2	14

I started with an arbitrary initial number of 20 gridblocks, and then I calculated Δ. Once the number of gridblocks and the size of the reservoir were adjusted to guarantee

an integer number of gridblocks in each and every direction, I proceeded to the inversion of the covariance matrix using both techniques.

Due to the huge CPU memory requirements of the numerical technique, I made the computer runs in batch mode. The maximum number of gridblocks I could use was limited to 5,000 due to my upper limit of 2 Giga bytes. It took the mainframe computer

Table 6.2 – Summary of 30 runs made to compare the efficiency of the numerical and the analytical techniques.

Run	M	Nx	Ny	Nz	Δt_N, seconds	Δt_A, seconds	Δx, ft	Δy, ft	Δz, ft
1	24	3	4	2	0.02	-	22.68	20.41	2.27
2	30	3	5	2	0.04	-	19.39	17.45	1.94
3	40	4	5	2	0.08	-	16.49	14.84	1.65
4	48	4	6	2	0.14	-	15.23	13.71	1.52
5	72	4	6	3	0.43	-	14.80	13.32	1.48
6	84	4	7	3	0.67	-	12.85	11.56	1.28
7	105	5	7	3	1.28	-	12.48	11.23	1.25
8	120	5	8	3	1.88	0.01	11.07	9.96	1.11
9	192	6	8	4	7.25	0.01	9.80	8.82	0.98
10	216	6	9	4	10.16	0.01	9.50	8.55	0.95
11	240	6	10	4	13.75	0.01	8.41	7.57	0.84
12	280	7	10	4	21.42	0.01	8.16	7.34	0.82
13	308	7	11	4	28.16	0.02	7.67	6.90	0.77
14	385	7	11	5	53.46	0.03	7.43	6.69	0.74
15	480	8	12	5	100.73	0.05	6.98	6.28	0.70
16	520	8	13	5	126.77	0.06	6.35	5.72	0.64
17	585	9	13	5	177.81	0.07	6.16	5.54	0.62
18	756	9	14	6	371.43	0.11	5.96	5.37	0.60
19	900	10	15	6	612.90	0.17	5.42	4.88	0.54
20	960	10	16	6	737.74	0.19	5.09	4.58	0.51
21	1232	11	16	7	1,510.49	0.31	4.93	4.44	0.49
22	1309	11	17	7	1,797.84	0.34	4.78	4.30	0.48
23	1512	12	18	7	2,720.26	0.45	4.49	4.04	0.45
24	1824	12	19	8	4,662.83	0.67	4.21	3.79	0.42
25	2080	13	20	8	6,799.84	0.86	4.08	3.67	0.41
26	2352	14	21	8	9,678.78	1.10	3.83	3.45	0.38
27	2646	14	21	9	13,575.68	1.42	3.71	3.34	0.37
28	2772	14	22	9	15,516.64	1.52	3.59	3.23	0.36
29	3105	15	23	9	21,494.44	1.94	3.48	3.13	0.35
30	3240	15	24	9	24,289.67	2.12	3.37	3.04	0.34

(K2) 29 hours to make a total of 30 runs. Of this total, only 11.48 seconds were required by the analytical technique.

Table 6.2 summarizes the results. The void spaces in the time columns are the result of the resolution of FORTRAN in terms of CPU time ($10\,msecs$). Δt_N and Δt_A are the CPU times of the numerical and the analytical techniques, respectively.

Fig. 6.1 is a log-log plot of CPU time as a function of the number of reservoir parameters using either technique.

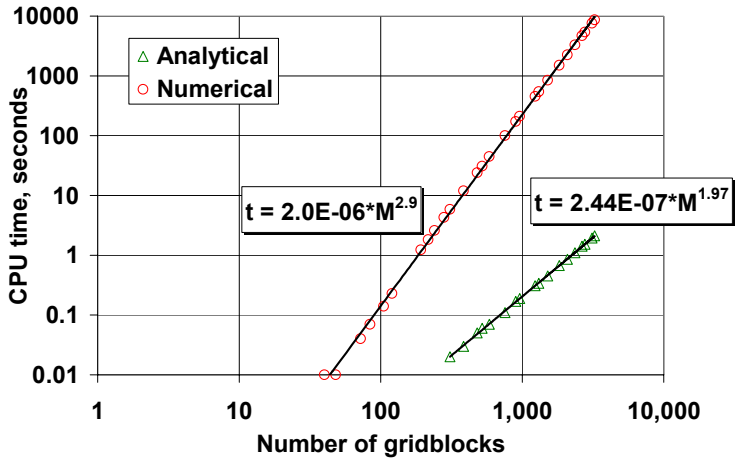


Fig. 6.1 – Pattern of growth of CPU time as a function of the number of reservoir parameters.

It is interesting to note that the analytical technique not only uses less CPU time but it also increases at a smaller rate. From the linear regression performed on the results shown in **Fig. 6.1**, the rates of growth of CPU time for both techniques can be summarized as follows:

a. Numerical technique:

$$t = C_1 M^{2.9} \qquad \text{.....(6.1)}$$

where $C_1 = 2.0 \times 10^{-6}$

b. Analytical technique:

$$t = C_2 M^{1.97} \quad \text{.....(6.2)}$$

where $C_2 = 2.44 \times 10^{-7}$

It can also be observed that the CPU time required by the analytical technique increases quadratically with the number of reservoir parameters (**Eq. 6.1**), as opposed to almost cubically in the case of the numerical technique (**Eq. 6.2**).

An interesting use of **Eq. 6.1** and **Eq. 6.2** would be to determine how refined the grid would have to be before it became prohibitive to use the numerical technique. For instance, it would take one day o invert the covariance matrix of a grid with 1,554 reservoir parameters. And a grid with 35,560 blocks would take one full year, assuming you could find a machine that could store the matrix. By contrast, the stencil technique would only take 7.78 seconds in the first case, and 4.9 minutes in the second. In addition, no huge arrays are necessary to invert the matrix using the analytical stencil; only a small 19-component one to store the stencil itself.

Scaling Behavior of Bayesian Approach Using the LSQR Algorithm

Fig. 6.2 compares the scaling behavior of the new Bayesian formulation to that of the Bayesian that uses Gauss-Newton and that of the deterministic. It can be observed that just like the deterministic formulation, it also follows a linear trend.

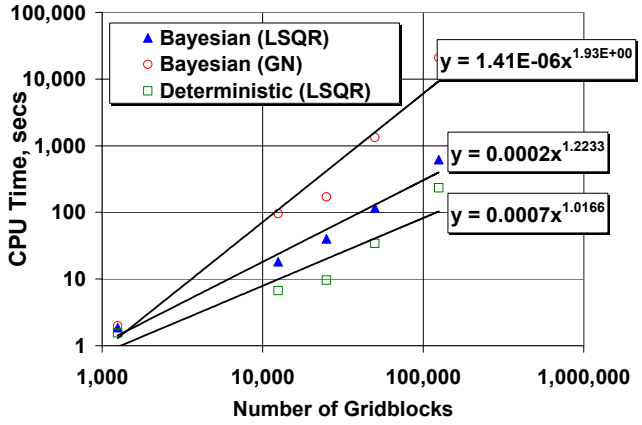


Fig. 6.2 –Scaling behavior of new Bayesian formulation is linear like that of the deterministic.

Summary

This chapter illustrates the computational advantage of using the analytically determined stencil over the numerical technique presented in Chapter IV.

In conclusion, the calculation increases quadratically with the number of reservoir parameters for the analytical technique. A cubic trend is observed in the case of the numerical technique.

This represents a major breakthrough, because it overcomes the most serious limitation of the reformulated Bayesian formulation presented in Chapter III.

CHAPTER VII

NUMERICAL-STENCIL TECHNIQUE TO APPROXIMATE THE SQUARE ROOT OF THE INVERSE OF THE COVARIANCE MATRIX

Chapter IV presented a technique to numerically calculate the square root of the inverse of the covariance matrix. Although the results yielded by this technique are very accurate, it becomes computationally prohibitive for applications involving a large number of unknowns, such as field-scale applications.

To alleviate this hurdle, Chapter V presents an alternative approximate solution in which the kernel of the square root of the inverse can be described using an analytically determined 3D stencil. Even though an excellent match between the numerical and the approximated results could be observed, the analytical technique has two major limitations:

- It was developed assuming an exponential model of the covariance function (**Eq. 5.1**), and
- The normalized grid sizes must be constant in all three directions of anisotropy (**Eq. 5.32**).

The following example illustrates the consequences of ignoring these two limitations.

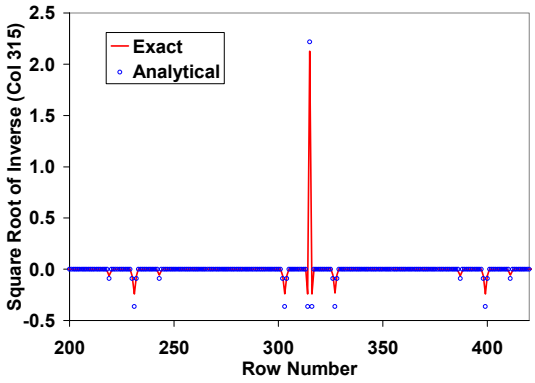
Ignoring the Limitations of the Analytical Technique

a. When the covariance model is other than exponential

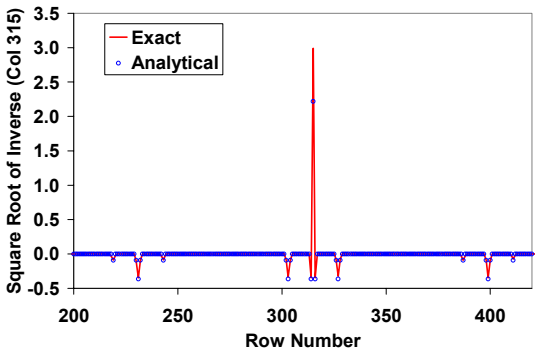
Table 7.1 shows the data used to generate the covariance matrix in this example.

Table 7.1 – Data used to generate covariance matrix (example 1).

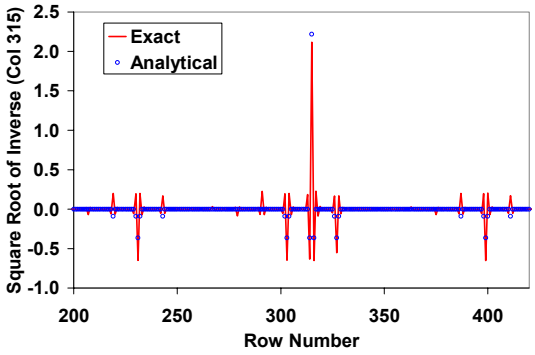
Model	σ^2	a	b	c	Δx	Δy	Δz	Nx	Ny	Nz
Exp	1.00	1.00	1.00	1.00	0.10	0.10	0.10	12	7	13
Sph	1.00	1.00	1.00	1.00	0.10	0.10	0.10	12	7	13
Gauss	1.00	1.00	1.00	1.00	0.50	0.50	0.50	12	7	13



(a)



(b)



(c)

Fig. 7.1 – The consequences of ignoring the constraint that the covariance model be exponential may be critical. (a) Exponential model, (b) spherical model, (c) Gaussian model.

Fig. 7.1 compares the approximation to the square root of the inverse using the analytical stencil presented in Chapter V for the exponential model of the covariance, and the matrices calculated using the numerical technique of Chapter IV for the exponential, spherical, and Gaussian covariance models.

Apparently, the analytical stencil provides a fairly good approximation for both the exponential and the spherical models (**Figs. 7.1a** and **7.1b**). The analytical stencil does a very poor job, however, when the covariance of the reservoir parameters is described with the Gaussian model (**Fig. 7.1c**).

b. When the ratio of the grid sizes to the ranges Δ, is not constant

Table 7.2 – Data used to generate covariance matrix (example 2).

σ^2	a	b	c	Δx	Δy	Δz	Nx	Ny	Nz
1.00	1.00	1.00	1.00	0.04	0.15	0.4	12	7	13

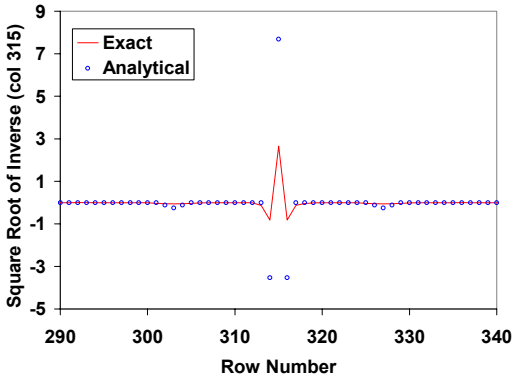


Fig. 7.2 – Consequences of ignoring the constraint of a constant normalized grid size may be critical.

As explained in Chapter V, the analytical technique for the determination of the stencil of the square root of the inverse was based on the use of convolution, along with the Fourier transform. This, in turn, called for the use of a normalized radial variable to transform the three normalized Cartesian coordinates. As a consequence, it was necessary to keep the changes in the three normalized Cartesian coordinates equal to a constant, as stated by **Eq. 5.32**. An example will illustrate the effect of ignoring this

constraint. **Table 7.2** shows the data used to generate the covariance matrix in this example.

Fig. 7.2 compares the exact square root of the inverse of the covariance matrix, to the approximation that would be obtained with the analytical approximation. Evidently, ignoring the constraint of a constant Δ led to a significant discrepancy.

Numerical-Stencil Technique

Because of the above limitations of the analytical technique, I developed a generalized technique to approximate the square root of the inverse of the covariance matrix. It incorporates the main features of the analytical and the numerical techniques. As illustrated later in this chapter, it preserves the savings in CPU time characteristic of the analytical technique.

This technique is based on two basic principles:

- The covariance matrix and the square root of its inverse can be constructed using their respective kernels;
- Regardless of the size of the matrices, the two kernels remain unchanged.

In Chapter IV, I succeeded in analytically calculating the kernel of the square root of the inverse for an exponential model, subject to the two constraints stated above. Analytical determination of this kernel for other covariance models (spherical and Gaussian) was not as straightforward. It is, however, relatively simple to determine the kernel of the square root of the inverse in the form of a stencil using a numerical technique.

Let C_M be a covariance matrix generated using the kernel of any covariance model. Then, using Sylvester's theorem (presented in Chapter IV)

$$C_M^{-1/2} = V^T \Lambda^{-1/2} V \quad \text{.....(7.1)}$$

where $C_M^{-1/2}$ is the square root of the inverse of the covariance matrix, V is a matrix whose columns are the eigenvectors of C_M, and $\Lambda^{-1/2}$ is a diagonal matrix of the eigenvalues of C_M raised to the $-1/2$ power.

Using the two principles stated above, the kernel of $C_M^{-1/2}$ can be approximated using a sufficiently small grid system, such that only the smallest distant terms are chopped off.

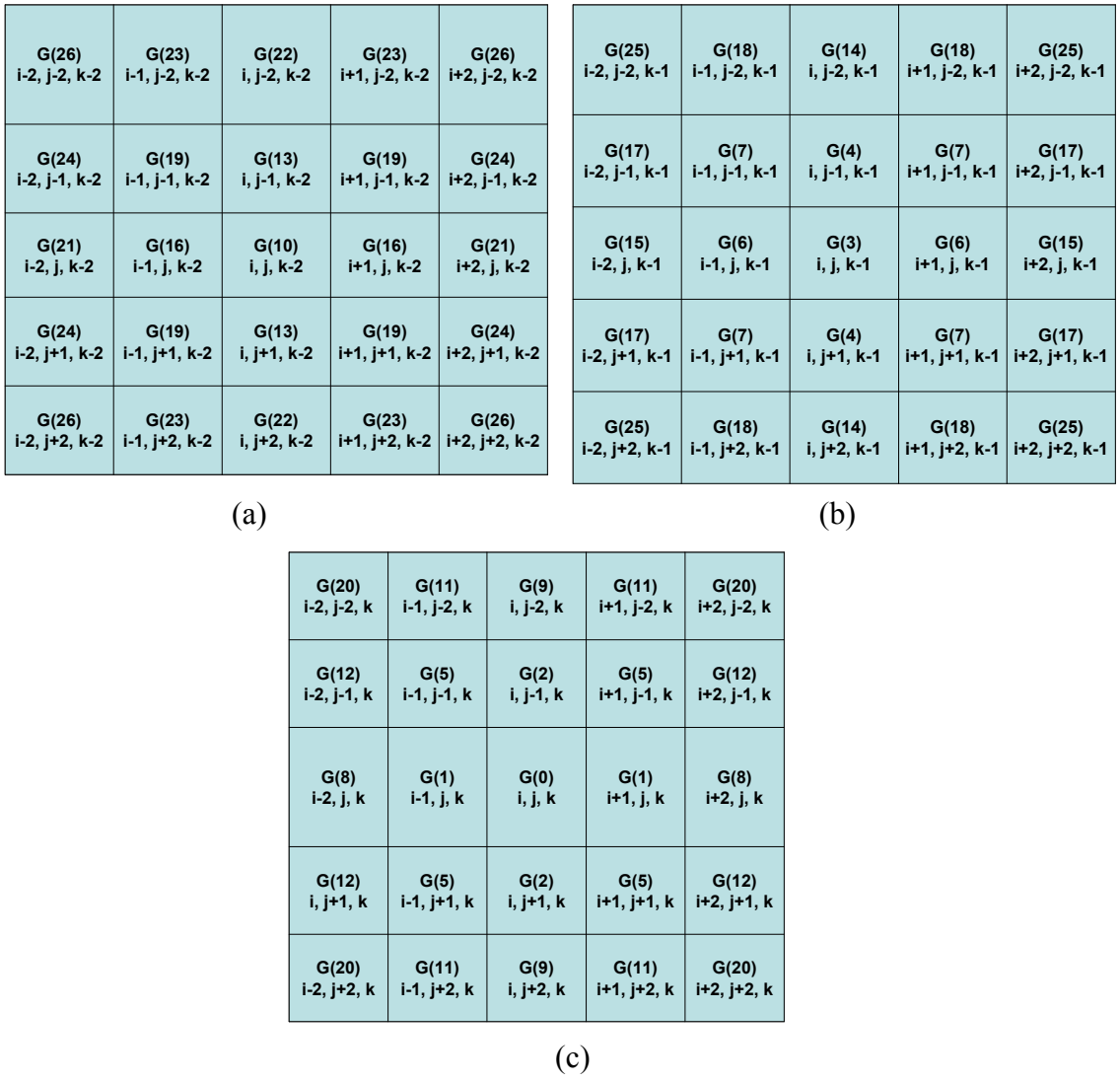


Fig. 7.3 –Numerically-determined stencil. (a) Layer $k-2$, (b) layer $k-1$, (c) layer k, (d) layer $k+1$, (e) layer $k+2$.

G(25) i-2, j-2, k+1	G(18) i-1, j-2, k+1	G(14) i, j-2, k+1	G(18) i+1, j-2, k+1	G(25) i+2, j-2, k+1
G(17) i-2, j-1, k11	G(7) i-1, j-1, k+1	G(4) i, j-1, k+1	G(7) i+1, j-1, k+1	G(17) i+2, j-1, k+1
G(15) i-2, j, k11	G(6) i-1, j, k+1	G(3) i, j, k+1	G(6) i+1, j, k+1	G(15) i+2, j, k+1
G(17) i-2, j+1, k+1	G(7) i-1, j+1, k+1	G(4) i, j+1, k+1	G(7) i+1, j+1, k+1	G(17) i+2, j+1, k+1
G(25) i-2, j+2, k11	G(18) i-1, j+2, k+1	G(14) i, j+2, k+1	G(18) i+1, j+2, k+1	G(25) i+2, j+2, k+1

(d)

G(26) i-2, j-2, k+2	G(23) i-1, j-2, k+2	G(22) i, j-2, k+2	G(23) i+1, j-2, k+2	G(26) i+2, j-2, k+2
G(24) i-2, j-1, k+2	G(19) i-1, j-1, k+2	G(13) i, j-1, k+2	G(19) i+1, j-1, k+2	G(24) i+2, j-1, k+2
G(21) i-2, j, k+2	G(16) i-1, j, k+2	G(10) i, j, k+2	G(16) i+1, j, k+2	G(21) i+2, j, k+2
G(24) i-2, j+1, k+2	G(19) i-1, j+1, k+2	G(13) i, j+1, k+2	G(19) i+1, j+1, k+2	G(24) i+2, j+1, k+2
G(26) i-2, j+2, k+2	G(23) i-1, j+2, k+2	G(22) i, j+2, k+2	G(23) i+1, j+2, k+2	G(26) i+2, j+2, k+2

(e)

Fig. 7.3 – (continued)

Based on the size of the stencil obtained for the analytical approximation in Chapter V, and some numerical experiments, I found that a $5\times5\times5$ stencil provided a good compromise between efficiency and accuracy. **Fig. 7.3** is a representation of the stencil terms that are to be determined numerically using **Eq. 7.1**.

The stencil has a total of 125 cells. However, because of symmetry, it is a combination of only 27 ($G(0)$ to $G(26)$) distinct terms. **Table 7.3** lists those 27 terms and their distance from the gridblock of interest, i, j, k.

Table 7.3 – Terms of Numerical Stencil.

Term of Numerical Stencil	Distance from Gridblock
G(0)	0
G(1)	Δx
G(2)	Δy
G(3)	Δz
G(4)	$\mathrm{Sqrt}(\Delta y^2+\Delta z^2)$
G(5)	$\mathrm{Sqrt}(\Delta x^2+\Delta y^2)$
G(6)	$\mathrm{Sqrt}(\Delta x^2+\Delta z^2)$
G(7)	$\mathrm{Sqrt}(\Delta x^2+\Delta y^2+\Delta z^2)$
G(8)	$2\Delta x$
G(9)	$2\Delta y$
G(10)	$2\Delta z$

Table 7.3 – (continued)	
G(11)	$\text{Sqrt}(\Delta x^2+(2\Delta y)^2)$
G(12)	$\text{Sqrt}((2\Delta x)^2+\Delta y^2)$
G(13)	$\text{Sqrt}(\Delta y^2+(2\Delta z)^2)$
G14)	$\text{Sqrt}((2\Delta y)^2+\Delta z^2)$
G(15)	$\text{Sqrt}((2\Delta x)^2+\Delta z^2)$
G(16)	$\text{Sqrt}(\Delta x^2+(2\Delta z)^2)$
G(17)	$\text{Sqrt}((2\Delta x)^2+\Delta y^2+\Delta z^2)$
G(18)	$\text{Sqrt}(\Delta x^2+(2\Delta y)^2+\Delta z^2)$
G(19)	$\text{Sqrt}(\Delta x^2+\Delta y^2+(2\Delta z)^2)$
G(20)	$\text{Sqrt}((2\Delta x)^2+(2\Delta y)^2)$
G(21)	$\text{Sqrt}((2\Delta x)^2+(2\Delta z)^2)$
G(22)	$\text{Sqrt}((2\Delta y)^2+(2\Delta z)^2)$
G(23)	$\text{Sqrt}(\Delta x^2+(2\Delta y)^2+(2\Delta z)^2)$
G(24)	$\text{Sqrt}((2\Delta x)^2+\Delta y^2+(2\Delta z)^2)$
G(25)	$\text{Sqrt}((2\Delta x)^2+(2\Delta y)^2+\Delta z^2)$
G(26)	$\text{Sqrt}((2\Delta x)^2+(2\Delta y)^2+(2\Delta z)^2)$

The procedure to calculate the 27 terms in **Table 7.3** can be summarized as follows

a. Construct the 125×125 covariance matrix corresponding to the $5\times5\times5$ grid illustrated in **Fig. 7.3**;

b. Calculate the square root of the inverse of the 125×125 covariance matrix, $C_M^{-1/2}$, using **Eq. 7.1**;

c. Read the values of $G(0)$ to $G(26)$ from any column of $C_M^{-1/2}$.

If the column in the middle (column 63) is selected—for convenience, the values of $G(0)$ to $G(26)$ would be read from the rows indicated in **Table 7.4**.

Table 7.4 – Determining terms of numerical stencil from column 63 of square root of inverse of covariance of $5\times5\times5$ grid.	
Numerical Stencil Term	**Row Number in Column 63**
G(0)	63
G(1)	62
G(2)	58
G(3)	38
G(4)	33
G(5)	57

Table 7.4 - Continued	
G(6)	37
G(7)	32
G(8)	61
G(9)	53
G(10)	13
G(11)	52
G(12)	56
G(13)	8
G14)	28
G(15)	36
G(16)	12
G(17)	31
G(18)	27
G(19)	7
G(20)	51
G(21)	11
G(22)	3
G(23)	2
G(24)	6
G(25)	26
G(26)	1

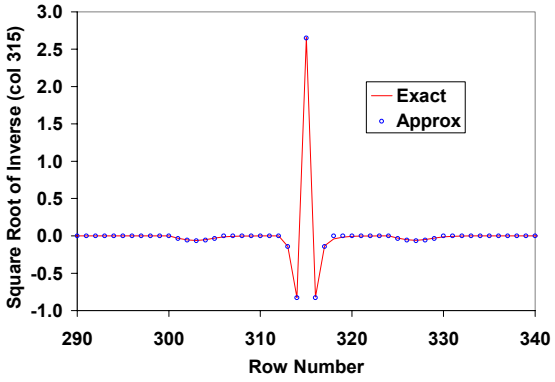
Illustrative Example

The following example will illustrate the effectiveness of the generalized technique to approximate the square root of the inverse of the covariance matrix.

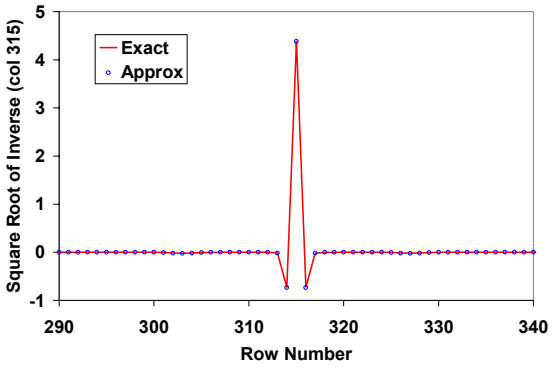
Table 7.5 shows the data used to generate the covariance matrix three different covariance models: exponential, spherical and Gaussian.

Table 7.5 – Data used to generate covariance matrix of illustrative example.

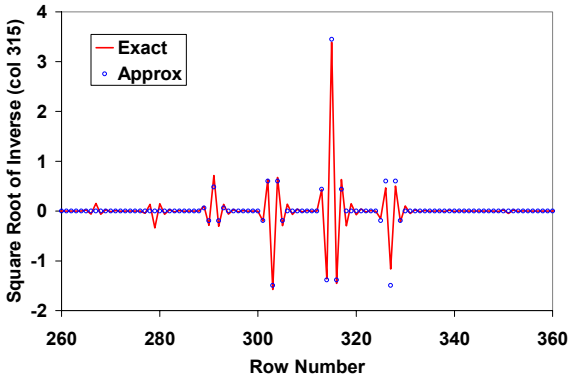
Model	σ^2	a	b	c	Δx	Δy	Δz	Nx	Ny	Nz
Exp	1.00	1.00	1.00	1.00	0.04	0.15	0.4	12	7	13
Sph	1.00	1.00	1.00	1.00	0.04	0.15	0.4	12	7	13
Gauss	1.00	1.00	1.00	1.00	0.43	0.41	0.5	12	7	13



(a)



(b)



(c)

Fig. 7.4 – The numerical approximation technique provides a fairly good match with exact square root of the inverse of the covariance matrix. (a) Exponential model, (b) spherical model, (c) Gaussian model.

Fig. 7.4 compares the approximation to the square root of the inverse using the generalized technique presented in this chapter, and the exact matrices calculated using the numerical technique of Chapter IV for the exponential, spherical, and Gaussian covariance models

As a matter of fact, the generalized technique provides an excellent approximation for the exponential, the spherical and the Gaussian models. Besides, the constraint imposed by the analytical technique—that the ratio of the grid sizes to the ranges in the three directions of anisotropy must be constant—has been removed.

Computational Advantage

The generalized technique to approximate the square root of the inverse of the covariance matrix has two features in common with the analytical technique presented in Chapter V:

- It uses a stencil to describe the kernel of the square root of the inverse;
- It is highly efficient because it reduces the number of mathematical operations to a minimum.

Fig. 7.5 compares the CPU time required to calculate the square root of the inverse using the numerical technique of Chapter IV and the approximation presented in this chapter.

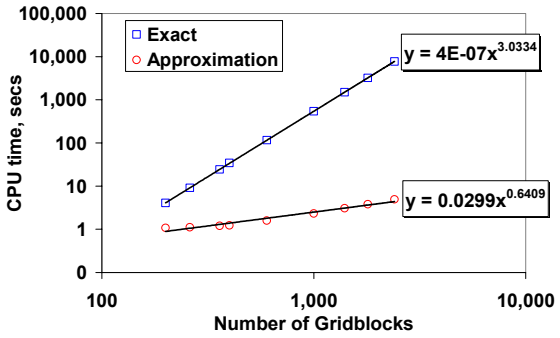


Fig. 7.5 – The numerical approximation preserves the computational advantage of the analytical approximation.

Comparison of **Fig. 7.5** to **Fig. 6.2** reveals that both approximation techniques offer an enormous computational advantage compared to the numerical technique.

Summary

The reformulated Bayesian formulation presented in Chapter III required the calculation of $C_M^{-1/2}$. From what has been covered in previous chapters, the following statements are relevant:

1. Numerical calculation of this matrix proved to be computationally prohibitive for large-scale problems.
2. The analytical approximation presented in Chapter V is subject to two constraints that limit its applicability.

This chapter presents a technique similar to that that uses an analytically determined stencil. The only difference is that the stencil is determined numerically using a technique based on Sylvester's theorem. The same theorem used in Chapter IV to perform the numerical calculation.

The numerical-stencil technique preserves the computational advantage of the analytically-determined stencil. Additionally,

1. It gets rid of both limitations, and
2. It does not have the truncation error in the approximation of the second derivative of the Dirac delta function. However, there could be some truncation error from limiting the size of the stencil to $5 \times 5 \times 5$.

CHAPTER VIII

ASSESSMENT OF UNCERTAINTY: THE RANDOMIZED MAXIMUM LIKELIHOOD METHOD

The purpose of this chapter is to assess uncertainty in future reservoir performance. Uncertainty is usually evaluated from the simulated performance of a small number of reservoir models. Unfortunately, most of the methods for creating reservoir models conditional to production data are known to generate a distribution of realizations that is only approximately correct. The correctness of the approximations is unknown, although several investigations of the approximate algorithms have suggested that the distributions of realizations could be seriously deceptive.

Liu *et al*.[21] evaluate the ability of the various sampling methods to correctly assess the uncertainty in reservoir predictions by comparing the distribution of realizations with a standard distribution from a Markov chain Monte Carlo method. This study compares the ensemble of realizations from five sampling algorithms for a synthetic, one-dimensional, single-phase flow problem in order to establish the best algorithm under controlled conditions. The small test problem was chosen in order that a sufficiently large number of realizations could be generated from each method to ensure the statistical validity of the comparisons.

The methods evaluated belong to two types: those that are known to sample correctly, and those that are only approximately correct. In the first category, they consider the Rejection algorithm and a Markov Chain Monte Carlo algorithm. The three approximate methods include Linearization about the Maximum a Posteriori, Randomized Maximum Likelihood, and Pilot Point methods.

From this study, it appears that, of the methods considered, generating realizations using the Randomized Maximum Likelihood (RML) method is the only practical alternative that provides acceptable assessment of uncertainty.

Randomized Maximum Likelihood Method

Kitanidis[22] and Oliver, He, and Reynolds[23] proposed that unconditional realizations from a Gaussian random field could be used to generate realizations conditional to nonlinear data by a process of minimization. If the prior covariance of the reservoir model parameters and the variance of the observed data are known, samples can be generated in the following way:

1. Generate an unconditional realization of the reservoir model parameters,

$$m_u \leftarrow N\left[m_{pr}, C_M\right] \quad (8.1)$$

2. Generate a realization of the data,

$$d_u \leftarrow N\left[d_{obs}, C_D\right] \quad (8.2)$$

3. Compute the set of model variables, m, that minimizes the function:

$$S(m) = \frac{1}{2}(m - m_u)^T C_M^{-1}(m - m_u) + \frac{1}{2}\left[g(m) - d_u\right]^T C_D^{-1}\left[g(m) - d_u\right] \quad (8.3)$$

The minimization step is similar to the computation of the maximum a posteriori estimate, with the difference that the regularization is with respect to unconditional realizations of the model and the data instead of the prior model and the observed data.

Oliver, *et al*.[23] originally suggested that this method be used to generate trial states for a Markov chain Monte Carlo algorithm but, because the acceptance criterion was difficult to evaluate and the acceptance rate was very high (approximately 95% for a small highly nonlinear problem), they suggested that the acceptance test be ignored and all trials accepted. Because the method seeks to minimize the data mismatch and the distance from the unconditional realization, the realizations almost surely honor the data and appear to be from the correct distribution.

The procedure proposed by Oliver *et al*.[23] to ensure that the realizations that are generated are distributed correctly is to use the calibrated realizations as trial states in a Markov chain Monte Carlo (MCMC) method. However, in order to be able to use the MCMC method, we need to be able to calculate the probability of proposing the calibrated model. The state m_{cal} that is proposed is the result of calibrating the unconditioned realization to the unconditioned data (observed data plus noise) using **Eq. 8.3**. The joint probability density, $f(m_{us}, d_{us})$ of proposing (m_{us}, d_{us}) is easily calculated because m_{us} and d_{us} are independent random variables. Hence, for this problem,

$$f(m_{us}, d_{us}) \propto \exp\left\{-\frac{1}{2}(m_{us}-\mu)^T C_M^{-1}\frac{1}{2}(m_{us}-\mu)^T - \frac{1}{2}[d_{us}-d_{obs}]^T C_D^{-1}[d_{us}-d_{obs}]\right\} \quad \text{.....(8.4)}$$

The joint probability density, $h(m_{cal}, d_{us})$, of proposing (m_{cal}, d_{us}) can, theoretically be calculated if the functional relationship between (m_{us}, d_{us}) and (m_{cal}, d_{us}) is known.

In their procedure, they calculate m_{cal} using a Gauss-Newton method to find the minimum of **Eq. 8.3**, given m_{us} and d_{us}. Reversing the procedure, m_{us} can be solved for as a function of m_{cal} and d_{us}. Excluding the regions of the (m_{cal}, d_{us}) space that are inaccessible to the calibration routine, we obtain a unique one-to-one, invertible, relationship between (m_{us}, d_{us}) and (m_{cal}, d_{us}). The joint probability of proposing (m_{cal}, d_{us}) can then be calculated as follows[24]

$$h(m_{cal}, d_{us}) = f(m_{us}, d_{us})|J| \quad \text{.....(8.5)}$$

where J is the Jacobian of the transformation,

$$J = \left|\frac{\partial m_{us}}{\partial m_{cal}}\right| \quad \text{.....(8.6)}$$

The probability of proposing m_{cal} is found by integrating $h(m_{cal}, d_{us})$ over the data space

$$q(m_{cal}) = \int_D h(m_{cal}, d_{us}) dd_{us} \quad \text{.....(8.7)}$$

For most practical problems, evaluation of the integral in Eq. 8.7 is too difficult to attempt. The authors then present a one-dimensional example for which the calculation can be attempted, and then show an approximation that seems to work well under a fairly broad range of conditions.

If the probability of proposing a transition to state m_j is independent of the current state, Hasting's rule for the acceptance of a proposed transition from state m_i to state m_j can be written as

$$\alpha_{i,j} = \min\left(1, \frac{\pi_j q_i}{\pi_i q_j}\right) \quad \text{.....(8.8)}$$

q_j is the probability of proposing the conditioned model and depends only on the proposed state. The probability density of the conditioned model, π_j is

$$\pi_j \propto \exp\left\{-\frac{1}{2}(m_j - \mu)^T C_M^{-1}(m_j - \mu) - \frac{1}{2}\left[g(m_j) - d_{obs}\right]^T C_D^{-1}\left[g(m_j) - d_{obs}\right]\right\} \quad \text{.....(8.9)}$$

Note that the probability is not based on the quality of the match obtained in the minimization, but on the quality of the match to the prior model and the observed data.

Oliver, *et al*. [23] demonstrate that for linear problems all calibrated reservoir problems will be accepted by a Metropolis-Hastings algorithm. For small nonlinear problems, they observed that accepting all calibrated models resulted in a reasonable

approximation to the correct distribution. Then, they bring up the question of whether there is any reason to believe that it might be a valid method of sampling for large multivariate problems.

They start by considering the probability density for proposing calibrated models. First, new states m_{us} and d_{us} are proposed from the Gaussian prior distribution,

$$f(m_{us}, d_{us}) \propto \exp\left\{-\frac{1}{2}(m_{us}-\mu)^T C_M^{-1} \frac{1}{2}(m_{us}-\mu)^T - \frac{1}{2}[d_{us}-d_{obs}]^T C_D^{-1}[d_{us}-d_{obs}]\right\} \quad (8.10)$$

A calibrated model, m_{cal}, is generated from m_{us} and d_{us} by minimizing

$$O(m) = \frac{1}{2}(m-m_{us})^T C_M^{-1}(m-m_{us}) + \frac{1}{2}[g(m)-d_{us}]^T C_D^{-1}[g(m)-d_{us}] \quad (8.11)$$

with respect to m . If the minimization is "good," $O(m_{cal})$ will be relatively small and m_{cal} and $g(m_{cal})$ will be close to m_{us} and d_{us}, respectively. In this case, the meaning of "close" is with respect to the weighted L_2 norm. Let $\varepsilon = m_{cal} - m_{us}$ and $\eta = g(m_{cal}) - d_{us}$. In terms of ε and η the distribution from which states are proposed can be written as

$$f(d_{us}, m_{us}) \propto \exp\left\{-\frac{1}{2}(m_{cal}-\varepsilon-\mu)^T C_M^{-1}(m_{cal}-\varepsilon-\mu)\right.$$
$$\left. -\frac{1}{2}[g(m_{cal})-\eta-d_{obs}]^T C_D^{-1}[g(m_{cal}-\eta-d_{obs})]\right\} \quad (8.12)$$

where m_{cal}, ε and η must be thought of as functions of m_{us} and d_{us}. Reorganization of the terms results in an equivalent expression in which the first two terms of the argument of the exponential are independent of ε and η

$$f(d_{us}, m_{us}) \propto \exp\Big\{ -\frac{1}{2}(m_{cal}-\mu)^T C_M^{-1}(m_{cal}-\mu)$$
$$-\frac{1}{2}\left[g(m_{cal})-d_{obs}\right]^T C_D^{-1}\left[g(m_{cal}-d_{obs})\right]$$
$$+\varepsilon^T C_M^{-1}(m_{cal}-\mu)-\frac{1}{2}\varepsilon^T C_M^{-1}\varepsilon$$
$$+\eta^T C_D^{-1}\left[g(m_{cal})-d_{obs}\right]-\frac{1}{2}\eta^T C_D^{-1}\eta \ \Big\} \quad \text{.....(8.13)}$$

The probability density of proposing a state, m_{cal}, can be obtained by multiplying **Eq. 8.13** by the Jacobian of the transformation between m_{us} and m_{cal}, then integrating over the data space. We formally write this as

$$q(m_{cal}) \propto \exp\Big\{ -\frac{1}{2}(m_{cal}-\mu)^T C_M^{-1}(m_{cal}-\mu) \quad -\frac{1}{2}\left[g(m_{cal})-d_{obs}\right]^T C_D^{-1}\left[g(m_{cal}-d_{obs})\right] \Big\}$$
$$\int_{D_{us}} \exp\Big\{ \varepsilon^T C_M^{-1}(m_{cal}-\mu)-\frac{1}{2}\varepsilon^T C_M^{-1}\varepsilon + \eta^T C_D^{-1}\left[g(m_{cal})-d_{obs}\right]-\frac{1}{2}\eta^T C_D^{-1}\eta \ \Big\}|J|dd_{us} \text{.....(8.14)}$$

where now we must treat J, ε and η as functions of m_{cal} and d_{us}. The term outside the integral is the a posteriori probability density for the model. The Metropolis-Hastings acceptance criterion depends only on the ratio $\pi_j q_i / \pi_i q_j$. If this ratio is equal to one then the proposed transition to state *j* should be accepted. Direct computation of the ratio gives

$$\frac{\pi_j q_i}{\pi_i q_j} = \frac{\int_{D_{us}} \exp\Big\{ \varepsilon_i^T C_M^{-1}(m_i-\mu)-\frac{1}{2}\varepsilon_i^T C_M^{-1}\varepsilon_i + \eta_i^T C_D^{-1}\left[g(m_i)-d_{obs}\right]-\frac{1}{2}\eta_i^T C_D^{-1}\eta_i \Big\}|J_i|dd_{us}}{\int_{D_{us}} \exp\Big\{ \varepsilon_j^T C_M^{-1}(m_j-\mu)-\frac{1}{2}\varepsilon_j^T C_M^{-1}\varepsilon_j + \eta_j^T C_D^{-1}\left[g(m_j)-d_{obs}\right]-\frac{1}{2}\eta_j^T C_D^{-1}\eta_j \Big\}|J_j|dd_{us}}$$
..... (8.15)

It seems unlikely that useful bounds can be placed on this ratio for general nonlinear functions g but, because ε and η (which are small) occur in every term within the integral in **Eq. 8.15**, the authors claim that the ratio is of the order of one in the regions of interest.

Summary

Generating realizations of the permeability field drawn from a probability density function conditioned to inaccurate production data is difficult, because the problem is highly nonlinear. Inefficient methods that generate large numbers of rejected images must be ruled out as impractical because of the repeated need for reservoir flow simulation.

In this chapter, we present a two-step Markov chain Monte Carlo method for proposing transitions in the Metropolis-Hastings algorithm such that the resulting state has a high probability of acceptance. The first step is to propose an unconditional realization from a known probability distribution. This step could be carried out using any unconditional simulation technique so it is not limited to simple stochastic models. The second part of the proposed transition involves "history matching" of the unconditional simulation to production data that has noise added. The decision to accept or reject the resulting "history-matched" realization is made on the basis of the Metropolis-Hastings algorithm. Because, however, calculation of the acceptance criterion for calibrated models is very difficult, I present the approximate acceptance criterion proposed by Oliver, *et al*.[23] in which all proposed transitions are accepted. This provides a practical approach to uncertainty quantification during production data integration.

CHAPTER IX

FIELD ILLUSTRATION

Chapter VIII described the use of the Randomized Maximum Likelihood (RML) method, as summarized in **Eq. 8.1** to **Eq. 8.3**. As shown in Chapter II, optimization of the Bayesian objective function described in **Eq. 8.3** traditionally uses the Gauss-Newton algorithm which is not computationally efficient when compared to other solution techniques such as the LSQR algorithm. For that reason, in this chapter, instead of using **Eq. 8.3**, I use the efficient Bayesian formulation presented in Chapter III (**Eq. 3.19**) which can be optimized using the LSQR algorithm. I also illustrate the Randomized Maximum Likelihood method using a field example.

Field Description

We have applied the RML method along with the efficient Bayesian formulation to a CO_2 pilot project area in the Goldsmith San Andres Unit (GSAU), a dolomite formation in west Texas. The pilot area consists of nine inverted 5-spot patterns covering around 320 acres with average thickness of 100 ft and has over 50 years of production history prior to CO_2 project initiation in Dec 1996. I have used the waterflood production history prior to the CO_2 injection. **Fig. 9.1** shows the CO_2 pilot project site in the GSAU.

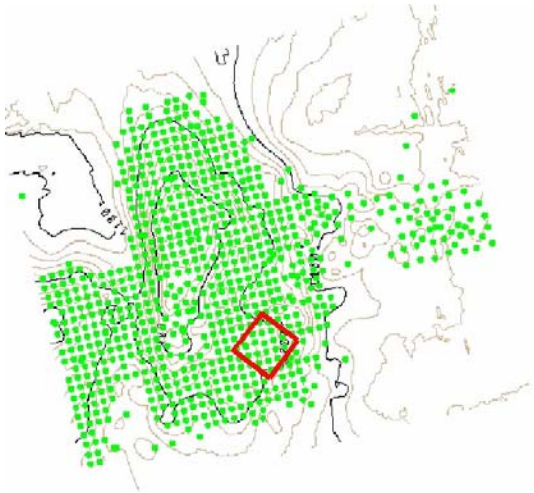


Fig. 9.1 – CO_2 pilot project site, Goldsmith field.

The extended study area is shown in **Fig. 9.2** with 11 water injectors and 31 producers.

The porosity field, **Fig. 9.3**, was generated from log data using sequential Gaussian simulation. It was not allowed to change during the integration.

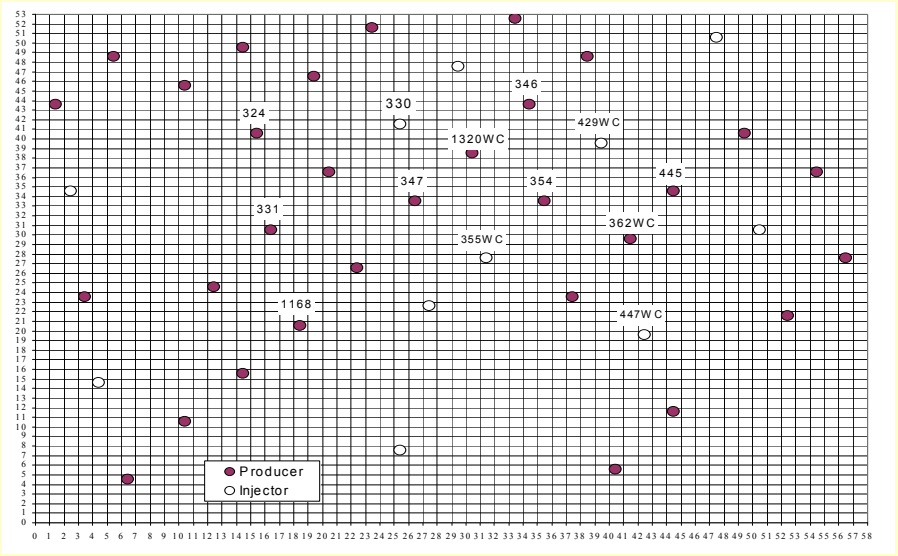


Fig. 9.2 – Extended study area, Goldsmith field.

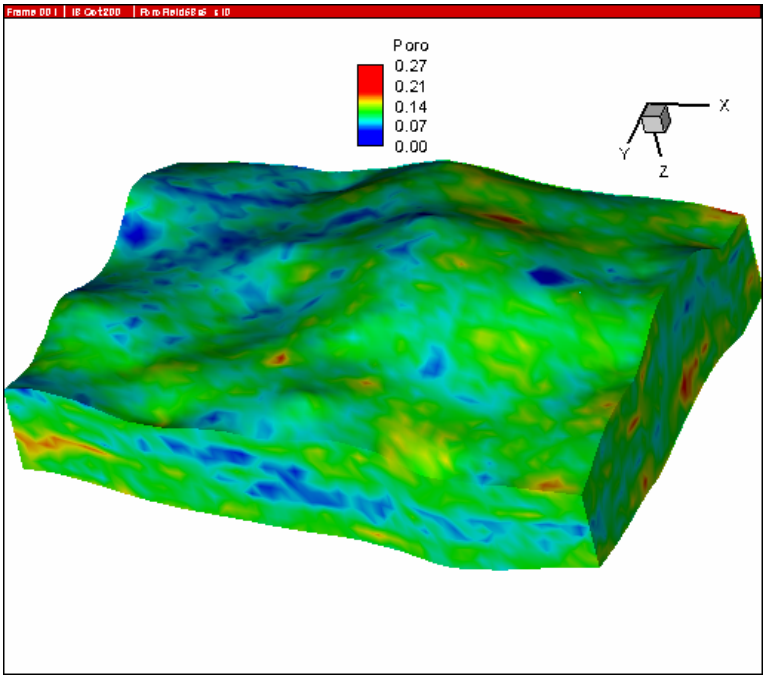


Fig. 9.3–Porosity field, Goldsmith field.

Permeability Fields

Five different realizations of the absolute permeability field were generated using a cloud transform[25].

A cloud transform basically consists of using the scatter or uncertainty in the relationship between porosity and permeability to generate permeability fields. Construction of a permeability fields involves the following steps:

1. Construct a probability field. This basically consists of assigning a cdf value to each gridblock. This is attained using a geostatistical model such as a sequential Gaussian simulation or a moving average technique.
2. For each gridblock,
    a. Pick the value of porosity,
    b. Assuming a porosity bin size, pick the corresponding values of permeability from the porosity-permeability relationship, and generate a permeability distribution,
    c. Sample the permeability distribution using the value of cdf corresponding to that particular gridblock.
3. Use the value of permeability obtained in **2c** as the unconditioned permeability of that particular gridblock,
4. Repeat steps **2** and **3** for all other gridblocks.

The resulting unconditioned permeability fields for five cases or realizations are shown in **Fig. 9.13a** to **Fig. 9.17a.**

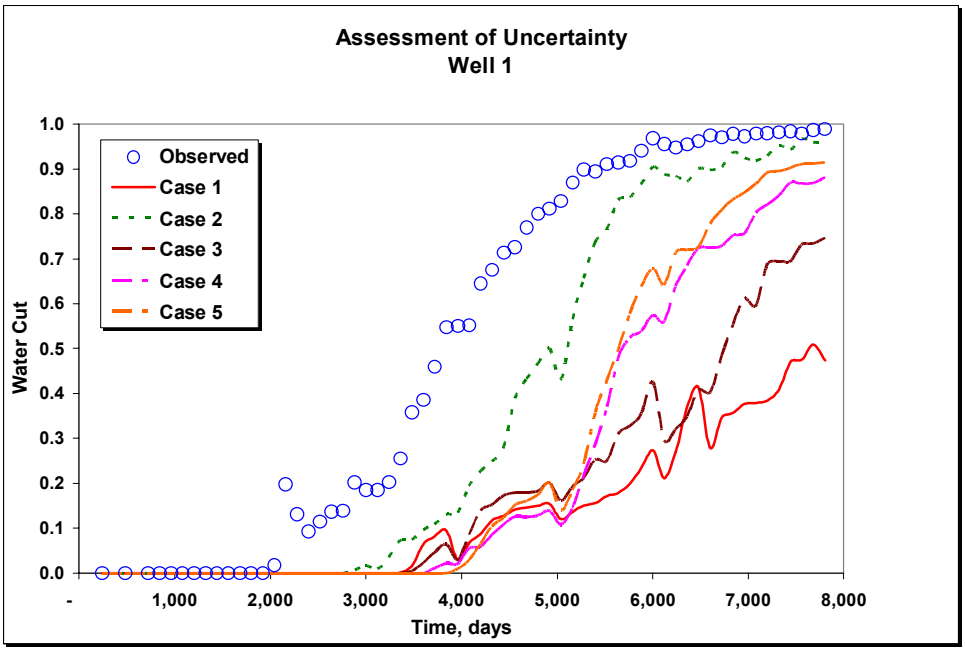
Forward runs were made using each of the unconditioned permeability fields and a streamline simulator. The unconditioned water cuts are shown in **Fig. 9.4** to **Fig. 9.12.**

Water Cut. Five unconditioned realizations of the water cut were generated by adding a randomly generated Gaussian error with a standard deviation of 0.03 to the observed water cut.

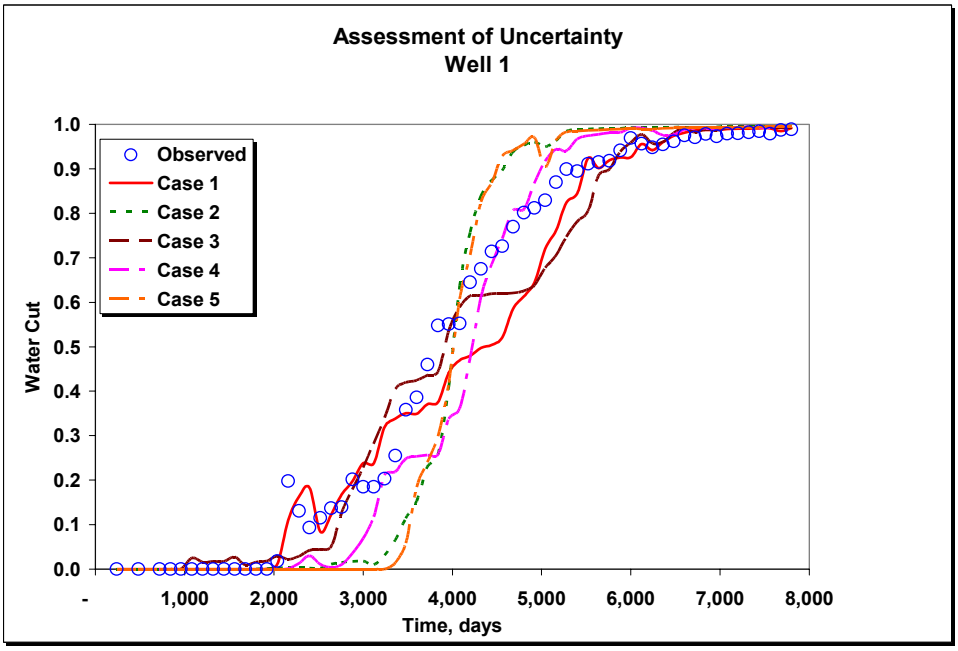
Data Integration

Using the entire 7800-day production history. To integrate the unconditioned permeability and water cut data, we used the efficient Bayesian formulation described in

Chapter III.

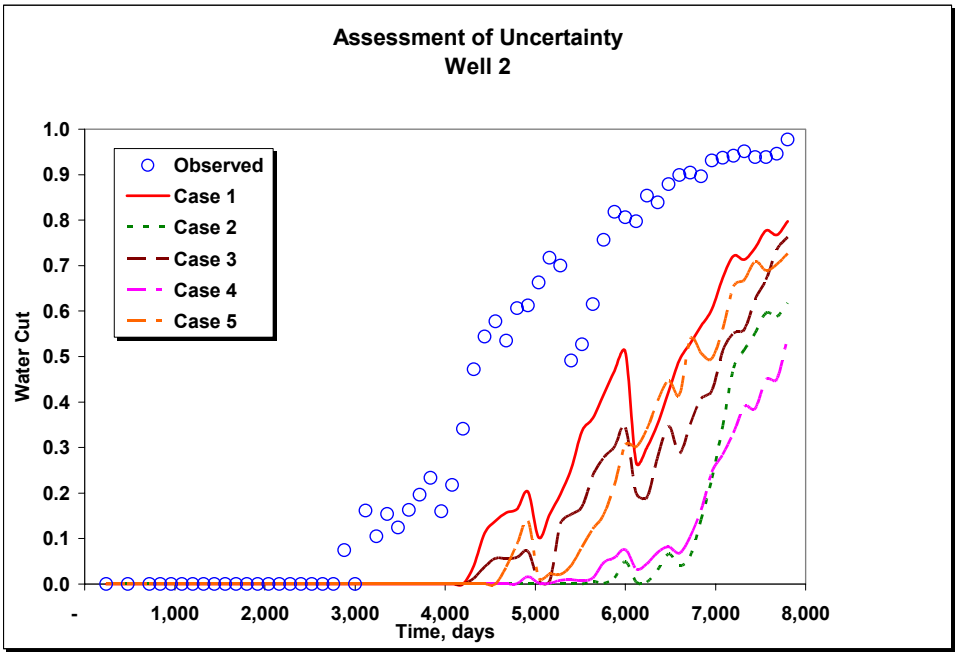


(a)

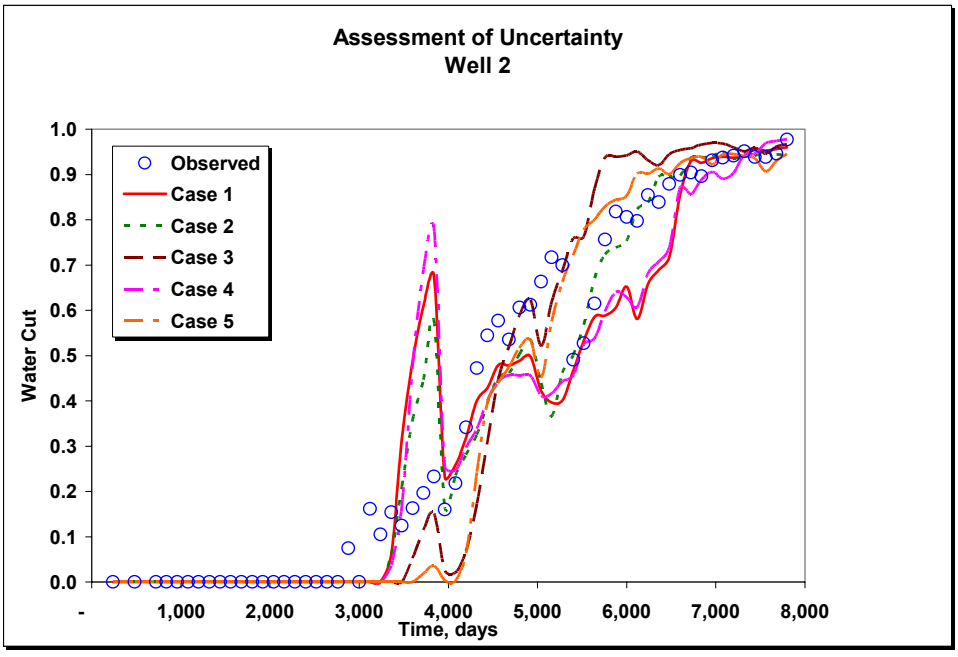


(b)

Fig 9.4 – Water cut, well 1. (a) Unconditioned, (b) conditioned.

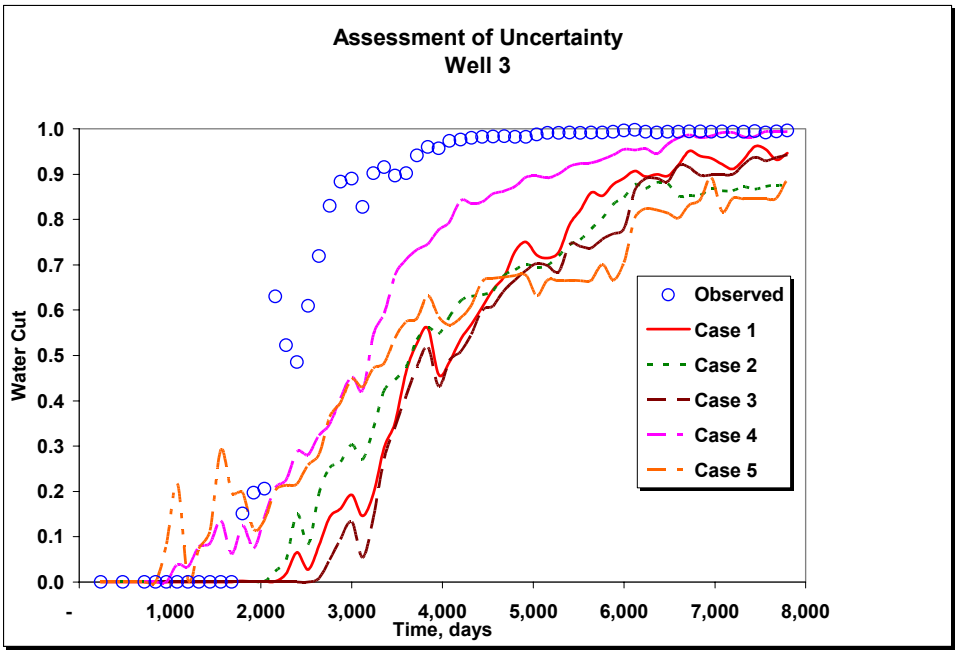


(a)

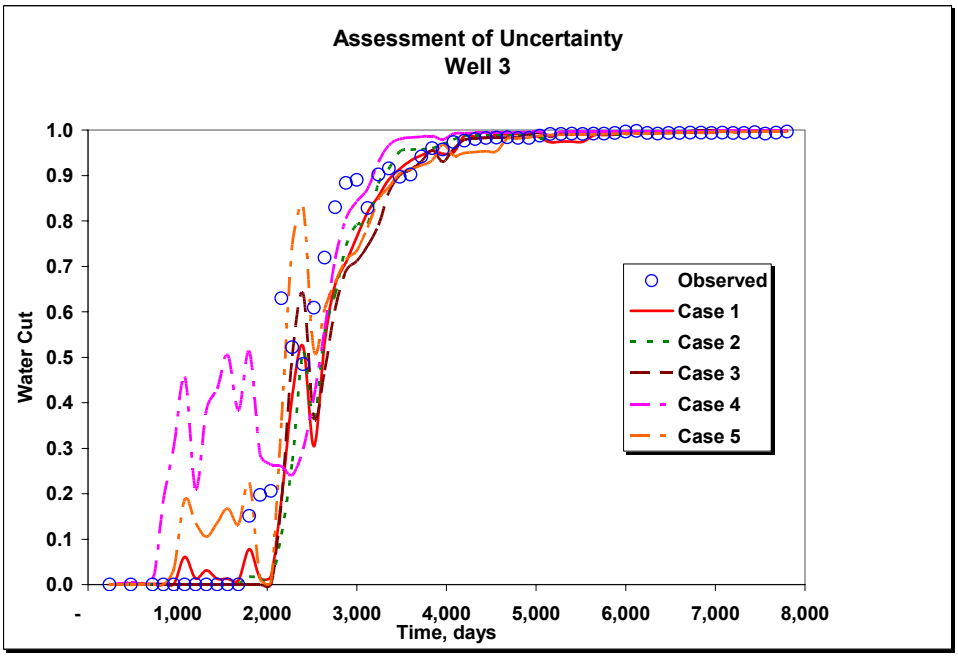


(b)

Fig 9.5 – Water cut, well 2. (a) Unconditioned, (b) conditioned.

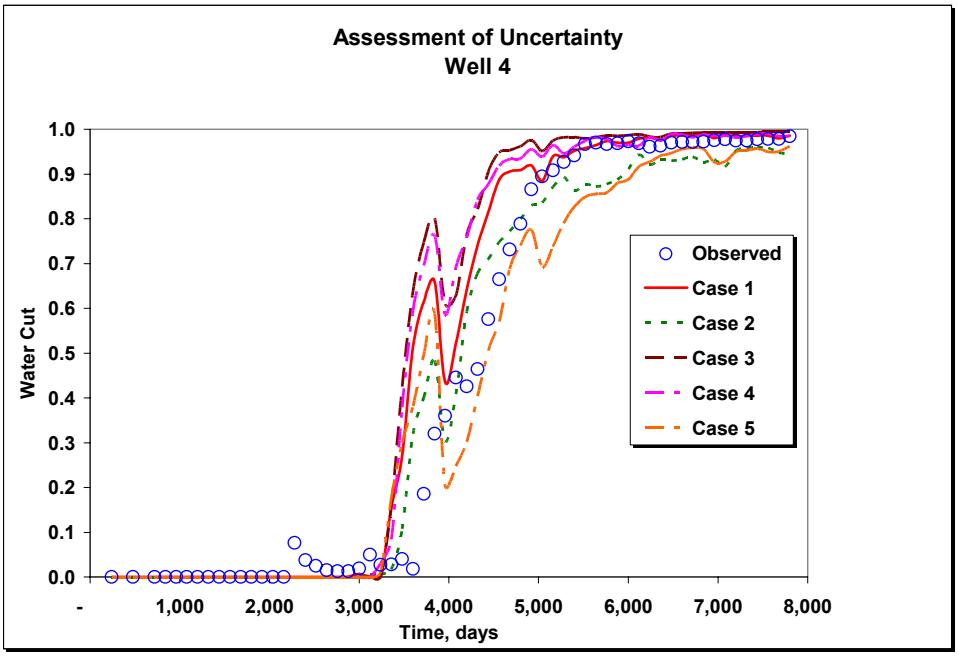


(a)

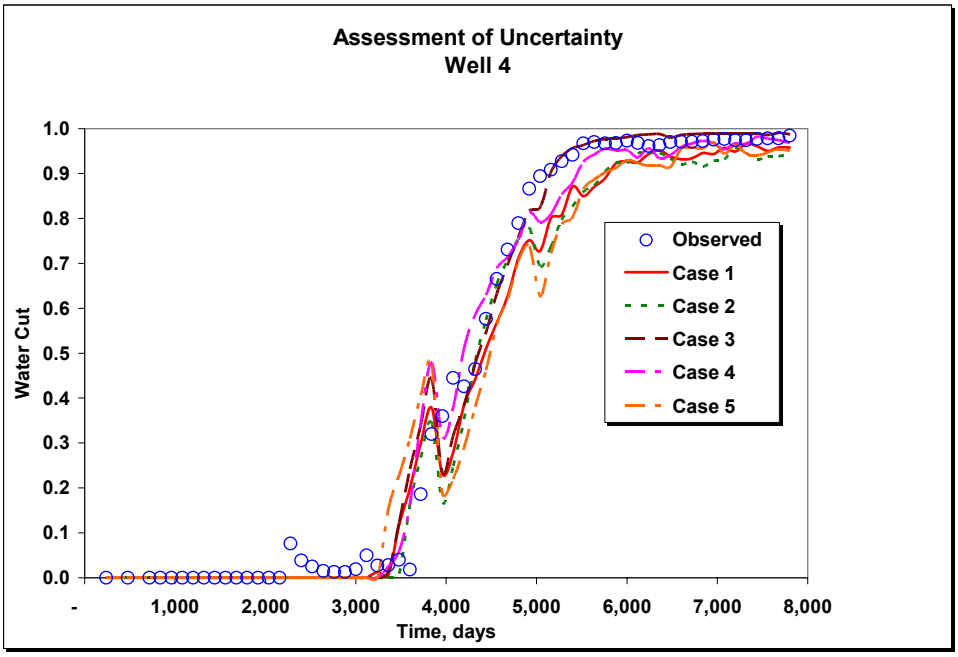


(b)

Fig 9.6 – Water cut, well 3. (a) Unconditioned, (b) conditioned.

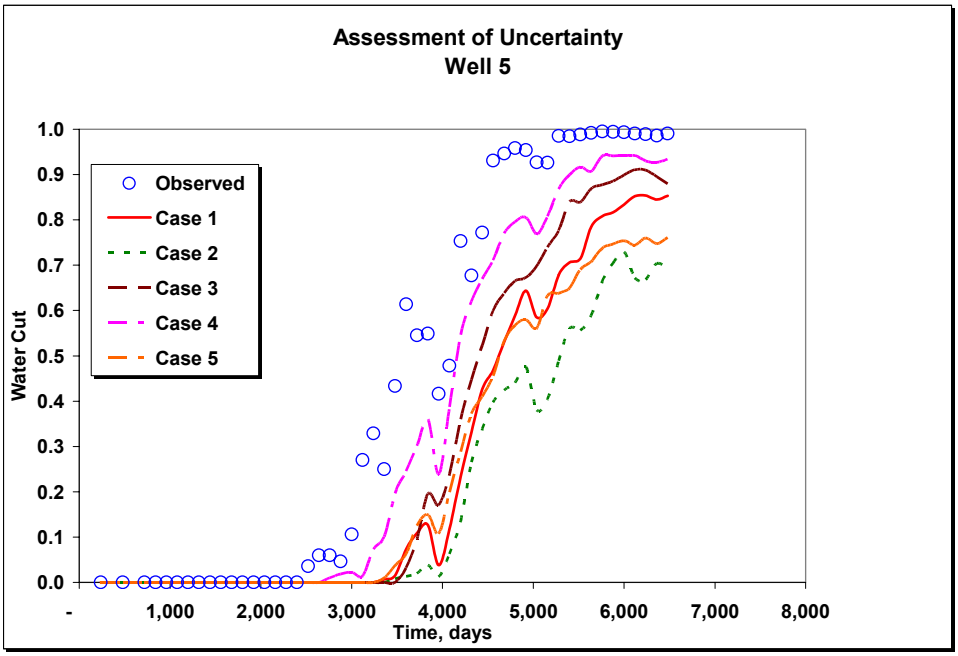


(a)

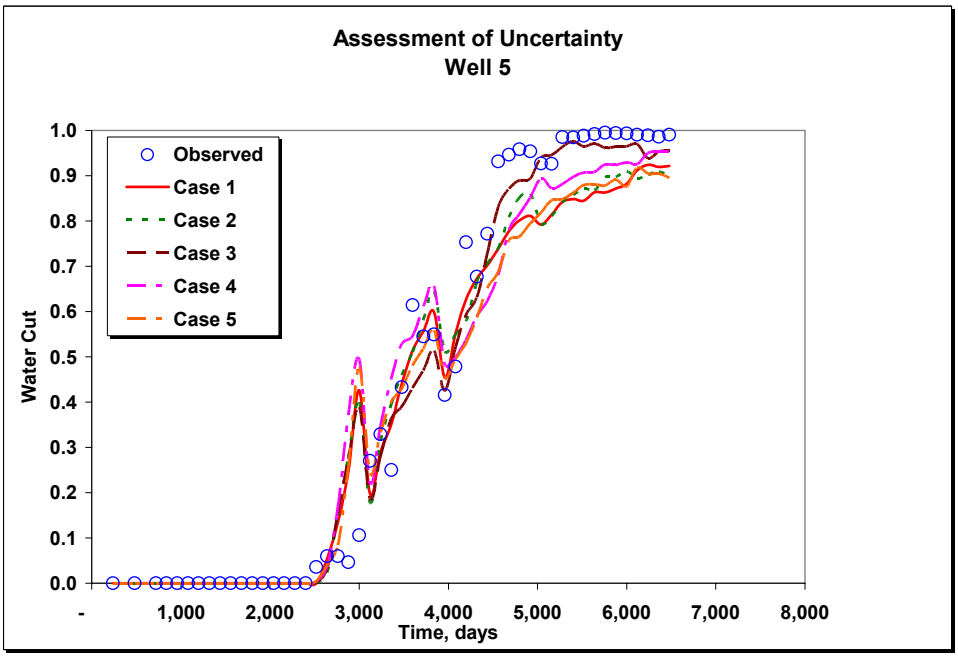


(b)

Fig 9.7 – Water cut, well 4. (a) Unconditioned, (b) conditioned.

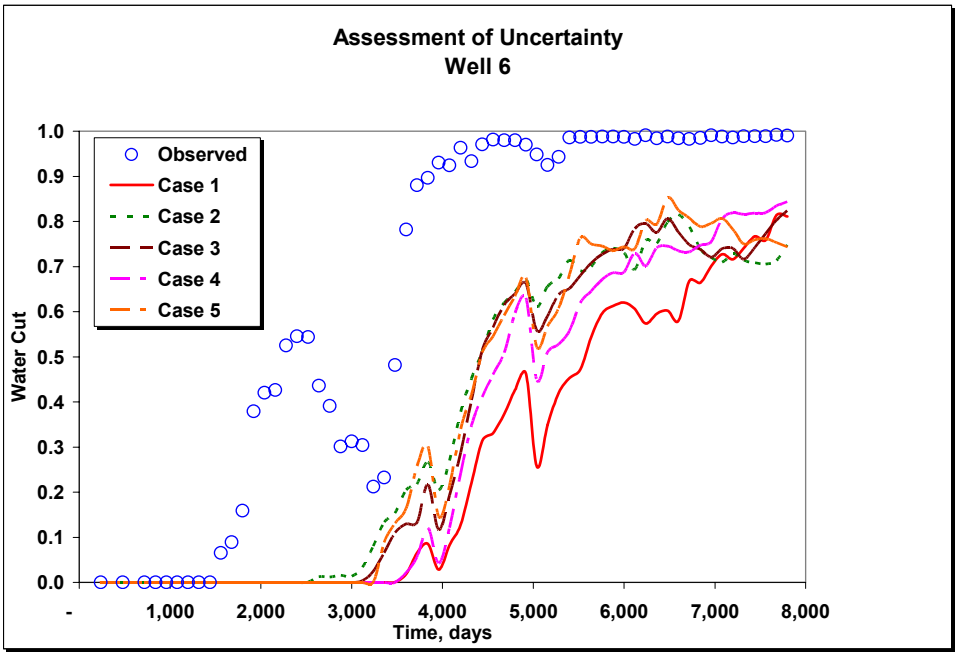


(a)

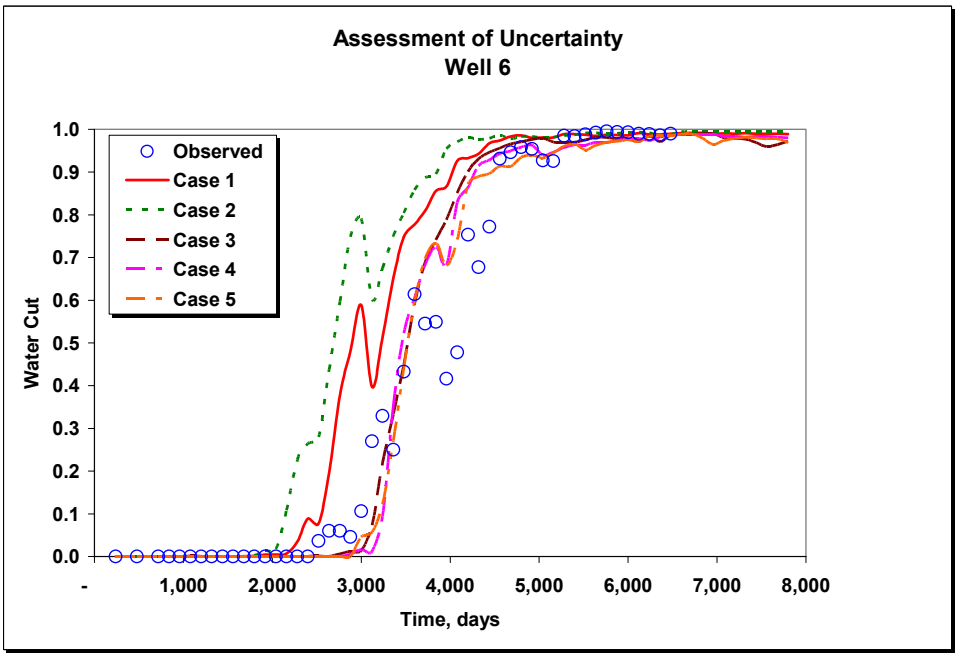


(b)

Fig 9.8 – Water cut, well 5. (a) Unconditioned, (b) conditioned.

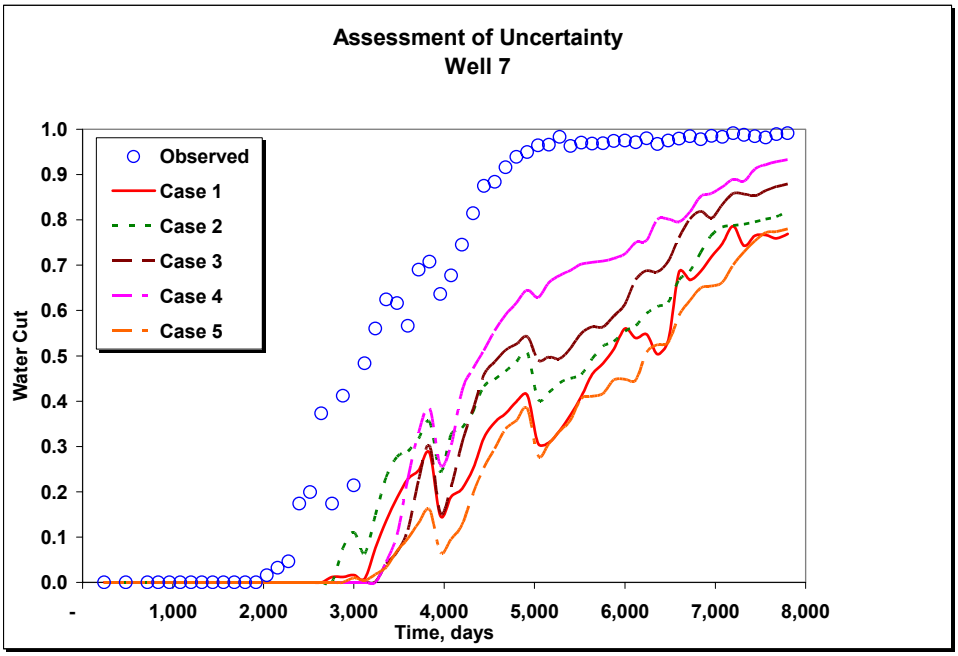


(a)

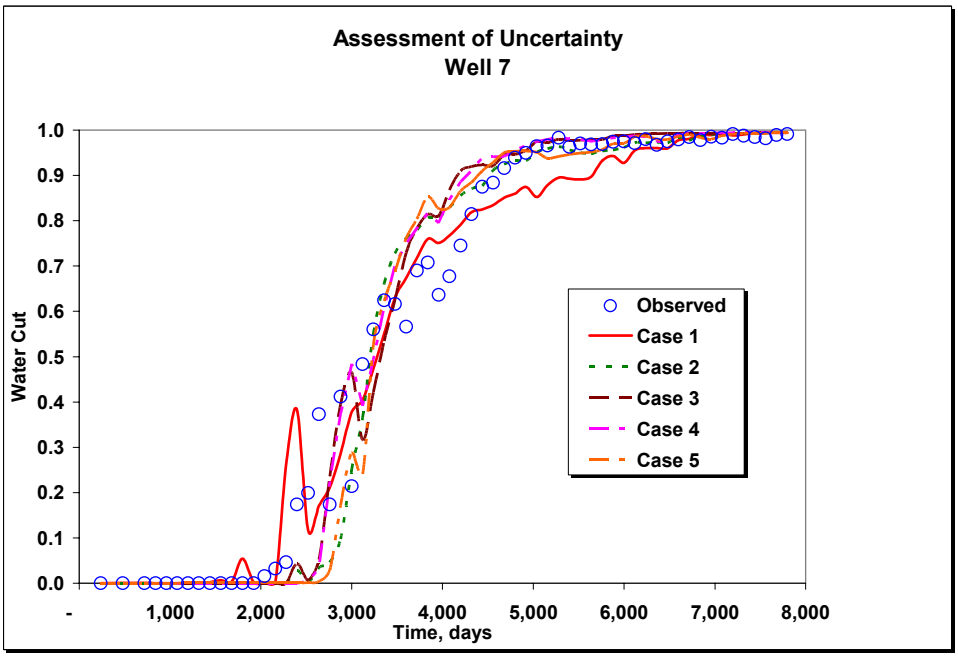


(b)

Fig 9.9 – Water cut, well 6. (a) Unconditioned, (b) conditioned.

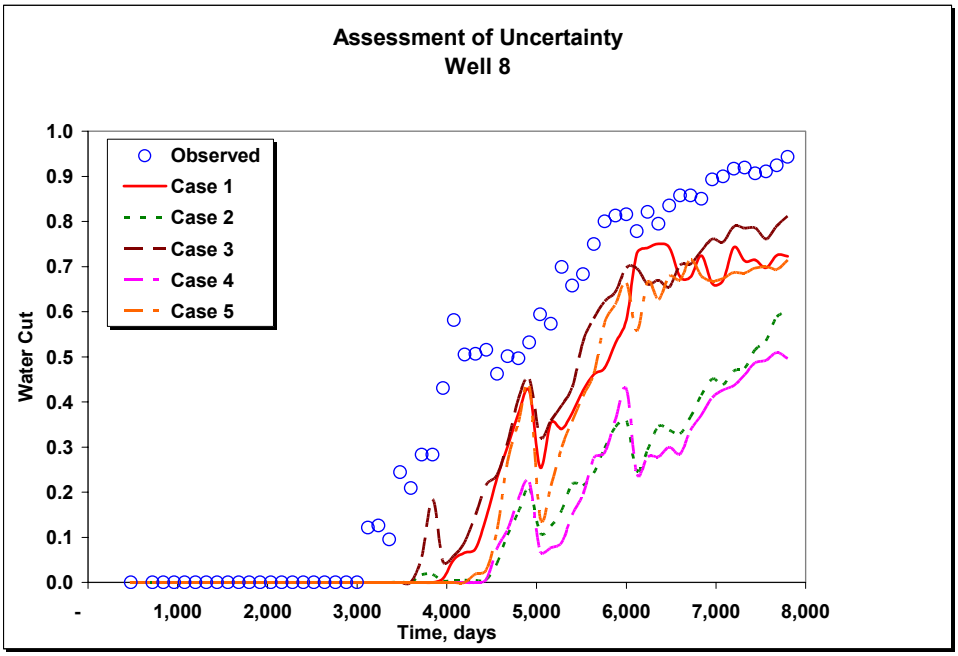


(a)

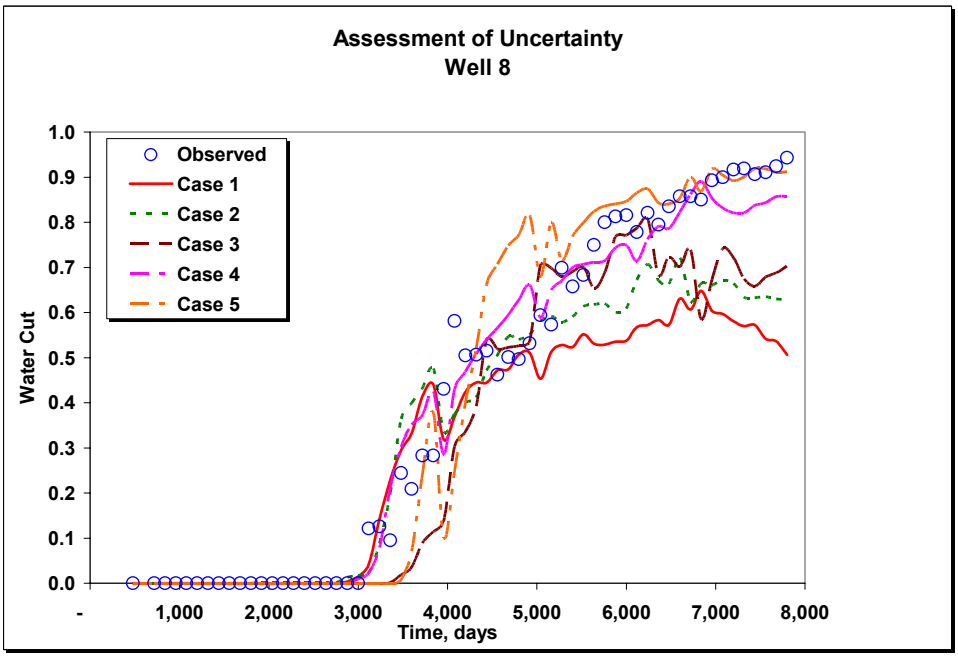


(b)

Fig 9.10 – Water cut, well 7. (a) Unconditioned, (b) conditioned.



(a)



(b)

Fig 9.11 – Water cut, well 8. (a) Unconditioned, (b) conditioned.

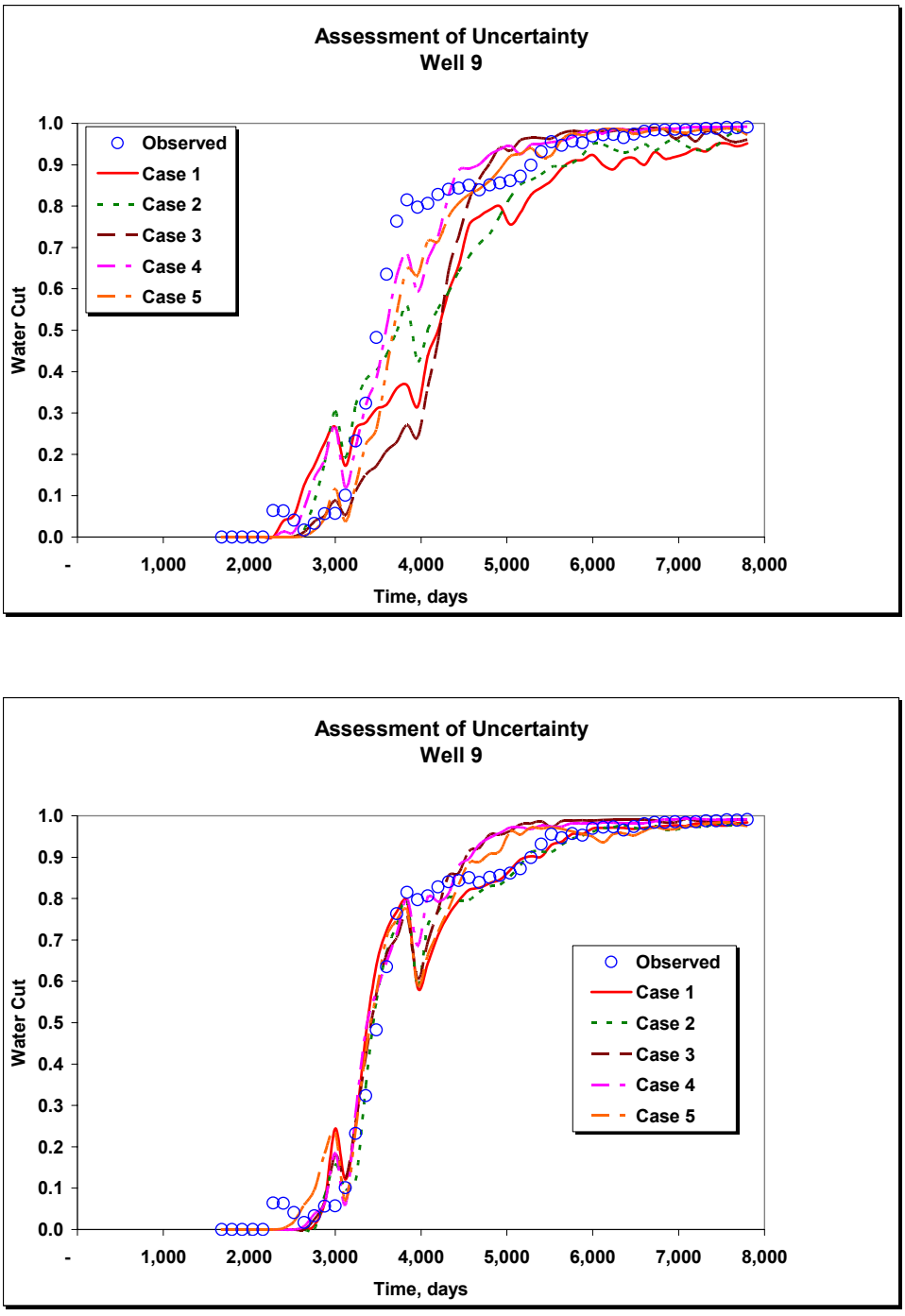
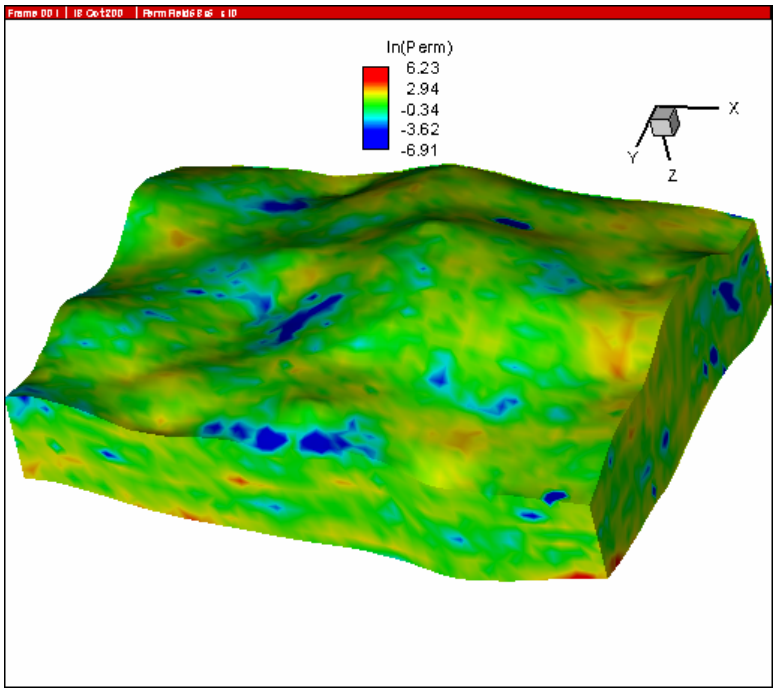
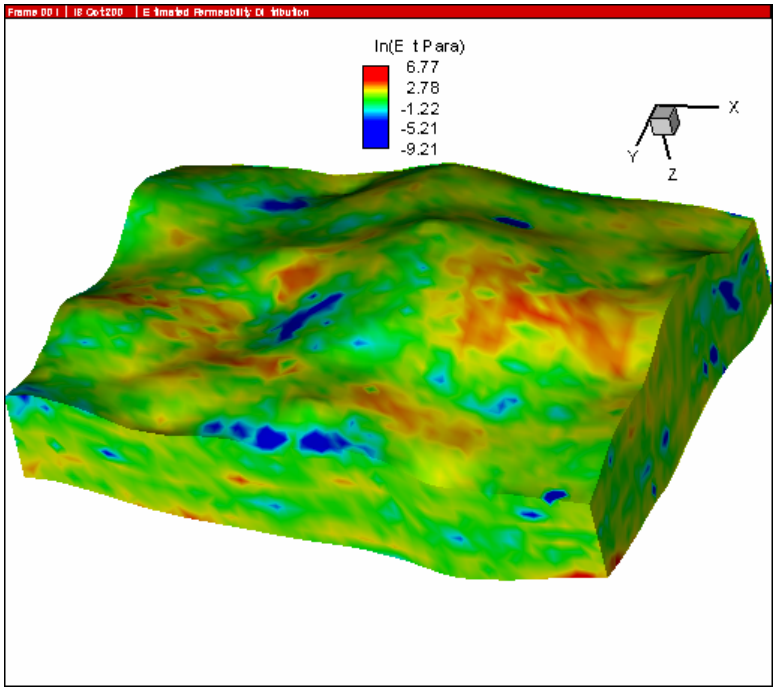


Fig 9.12 – Water cut, well 9. (a) Unconditioned, (b) conditioned.

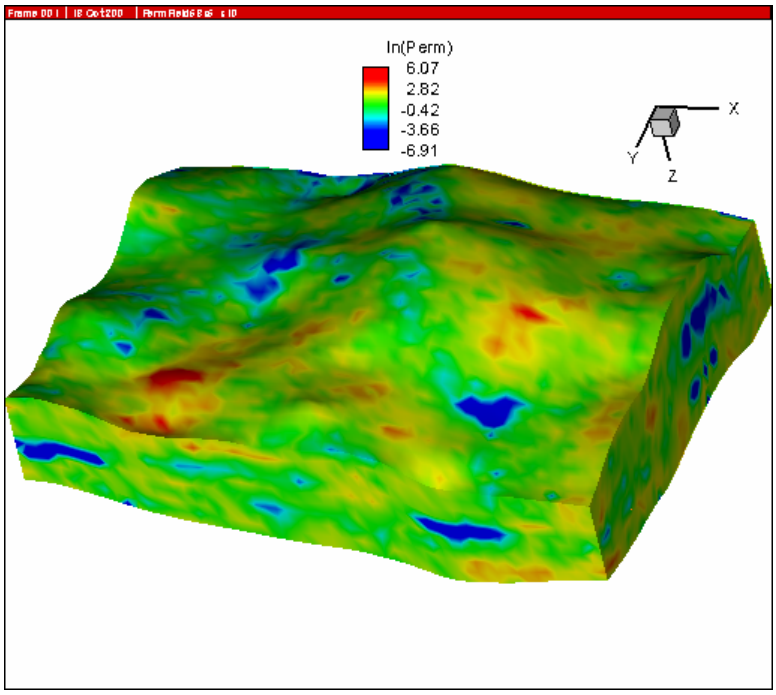


(a)

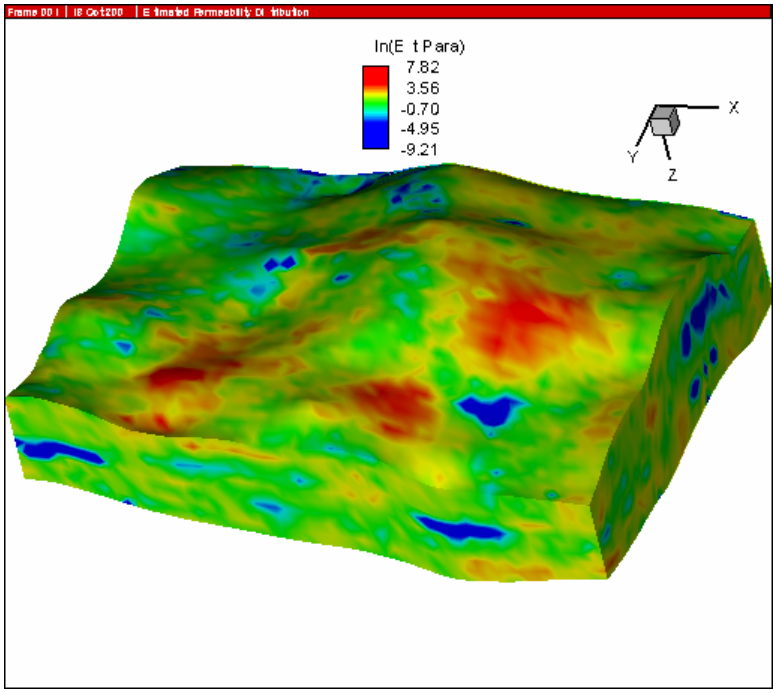


(b)

Fig. 9.13 – Permeability field, realization 1. (a) Unconditioned, (b) conditioned.

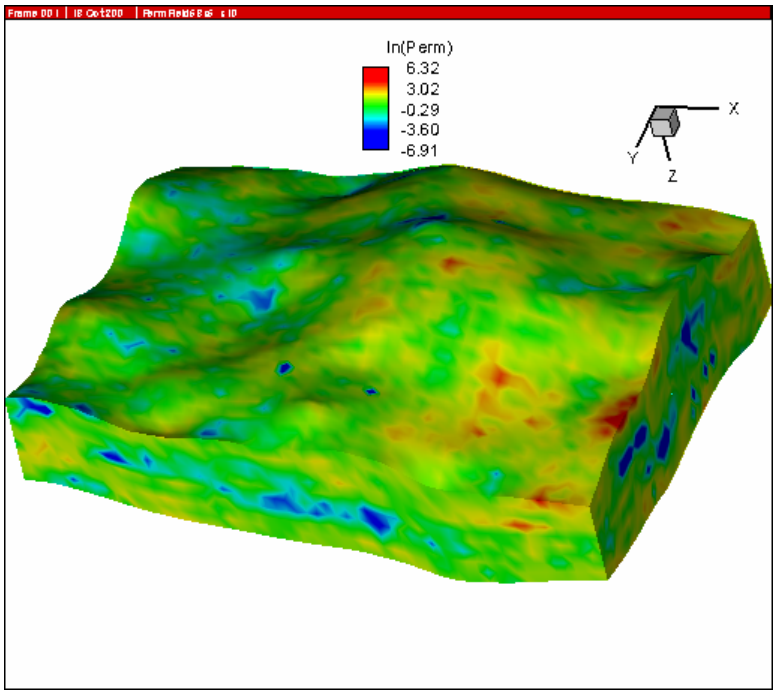


(a)

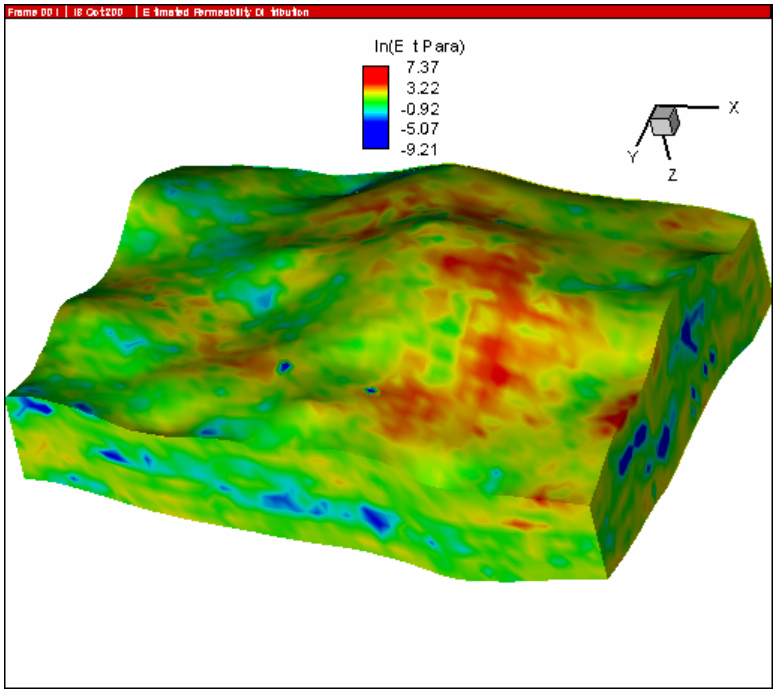


(b)

Fig. 9.14 – Permeability field, realization 2. (a) Unconditioned, (b) conditioned.

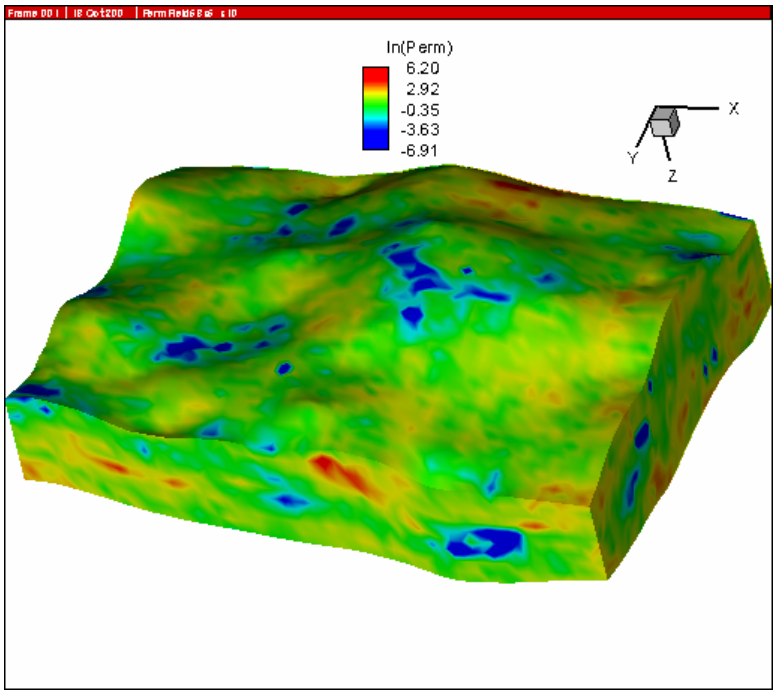


(a)

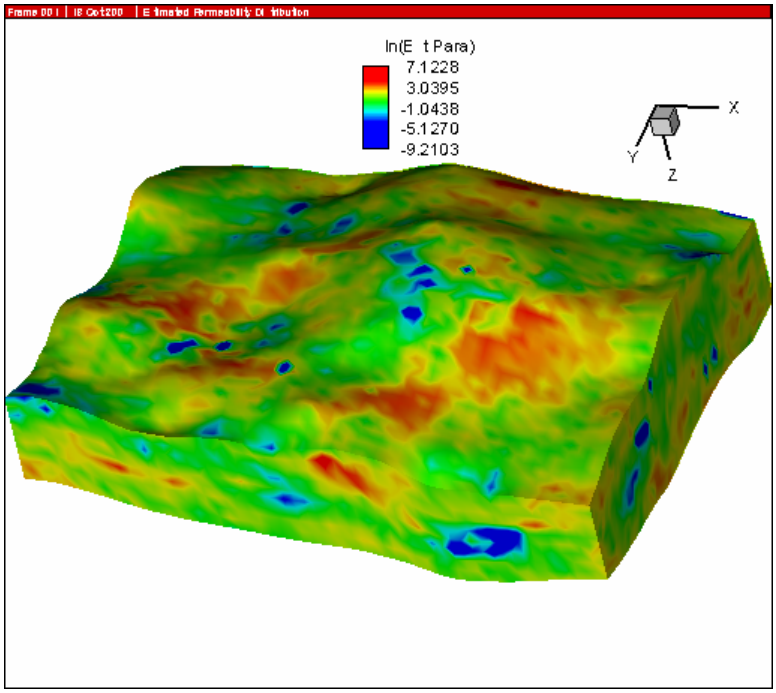


(b)

Fig. 9.15 – Permeability field, realization 3. (a) Unconditioned, (b) conditioned.



(a)



(b)

Fig. 9.16 – Permeability field, realization 4. (a) Unconditioned, (b) conditioned.

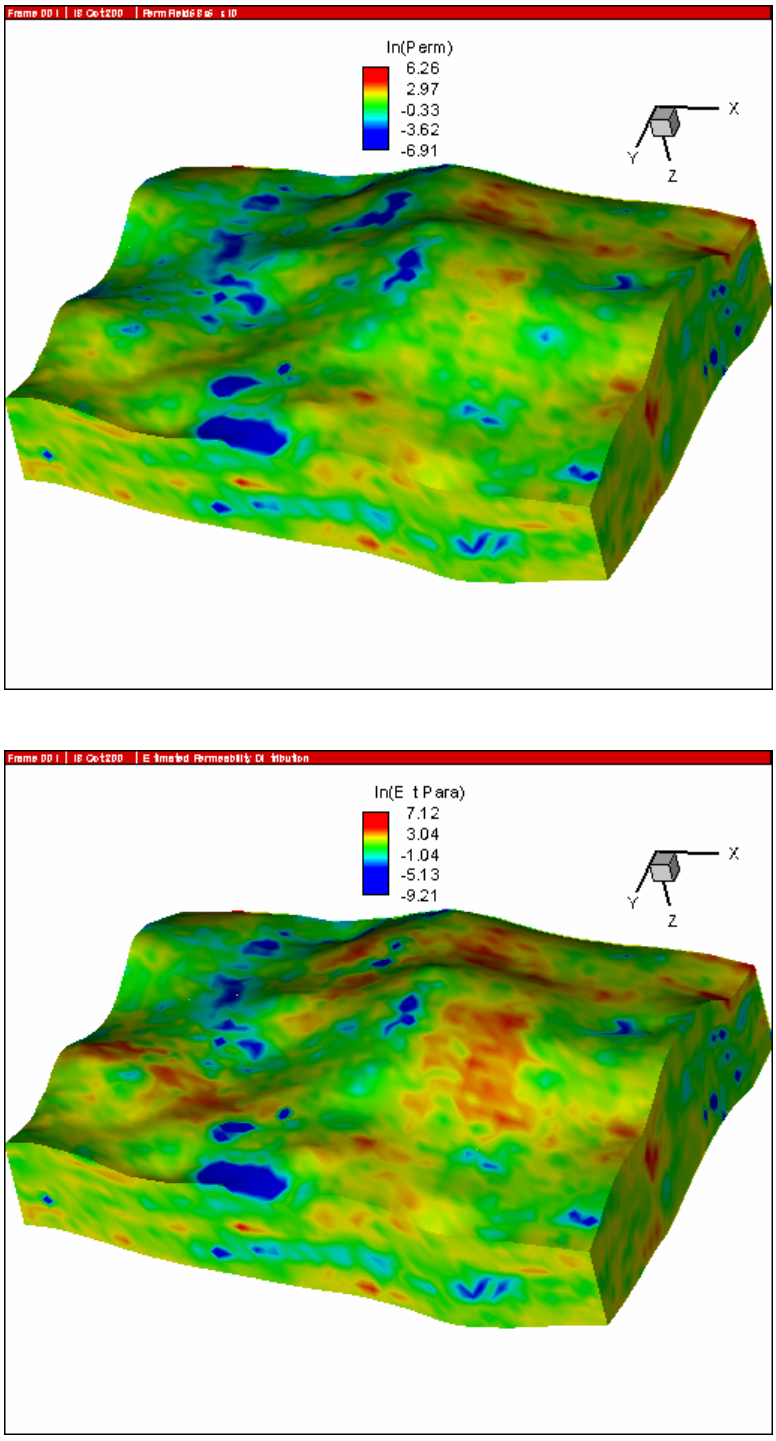


Fig. 9.17 – Permeability field, realization 5. (a) Unconditioned, (b) conditioned.

Water cut data was not used directly to calculate the data misfit term. Instead, the concept of generalized travel time[26] was implemented.

To simplify the calculation of the sensitivity coefficients, these were determined analytically using a streamline simulator. This technique required only one forward simulation to calculate the entire sensitivity matrix.

Fig. 9.4b to **Fig. 9.12b** show the conditioned water cuts for each of the five realizations. The corresponding five conditioned permeability fields are shown in **Fig. 9.13b** to **Fig. 9.17b**.

Upon comparison of the unconditioned to the conditioned ensemble of water cut realizations, two important observations can be made:

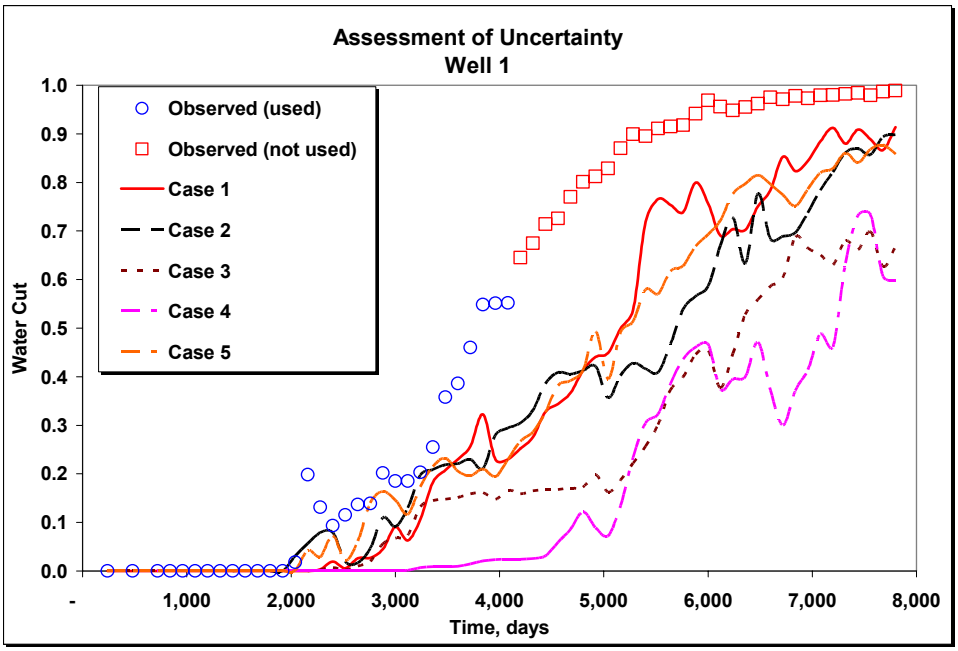
1. The match improves with the conditioning,
2. The spread of the realizations decreases as a result of the conditioning.

Using only one half of the production history. Given that the ultimate objective of this research is to be able to predict the uncertainty of predictions, our next two steps were

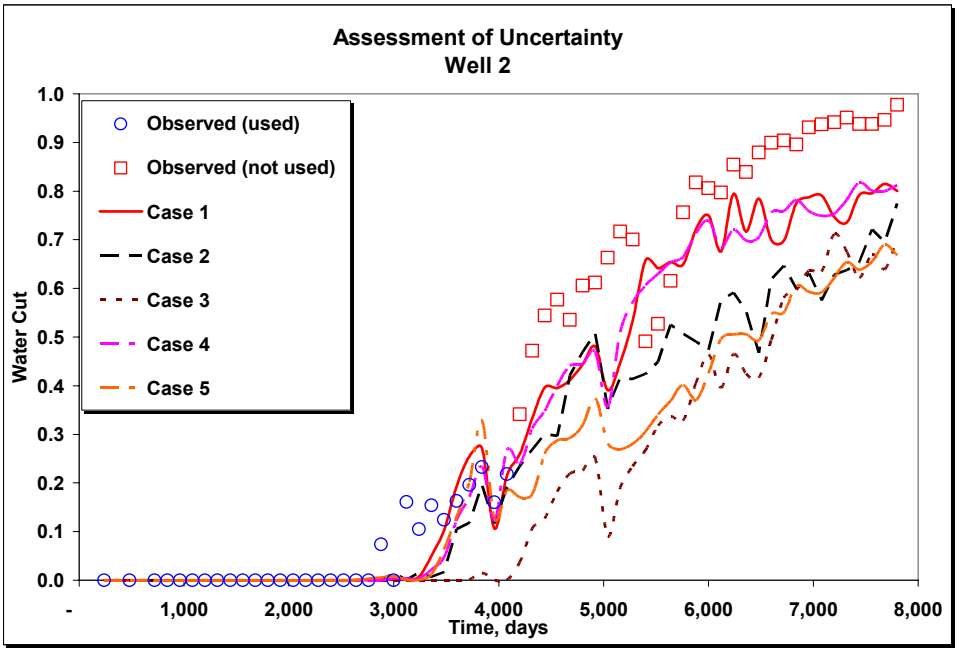
1. To perform a reservoir data integration using only one half of the available production history (4080 days), and
2. To predict production (water cut) performance using the conditioned permeability field.
3. To determine how uncertain those predictions are in comparison to the unconditioned (before history match) predictions.

Using the same unconditioned data as in the section above, a reservoir data integration was performed using production data up to 4,080 days. **Fig. 9.18** to **Fig. 9.26** show the history match and the predictions up to 7,800 days.

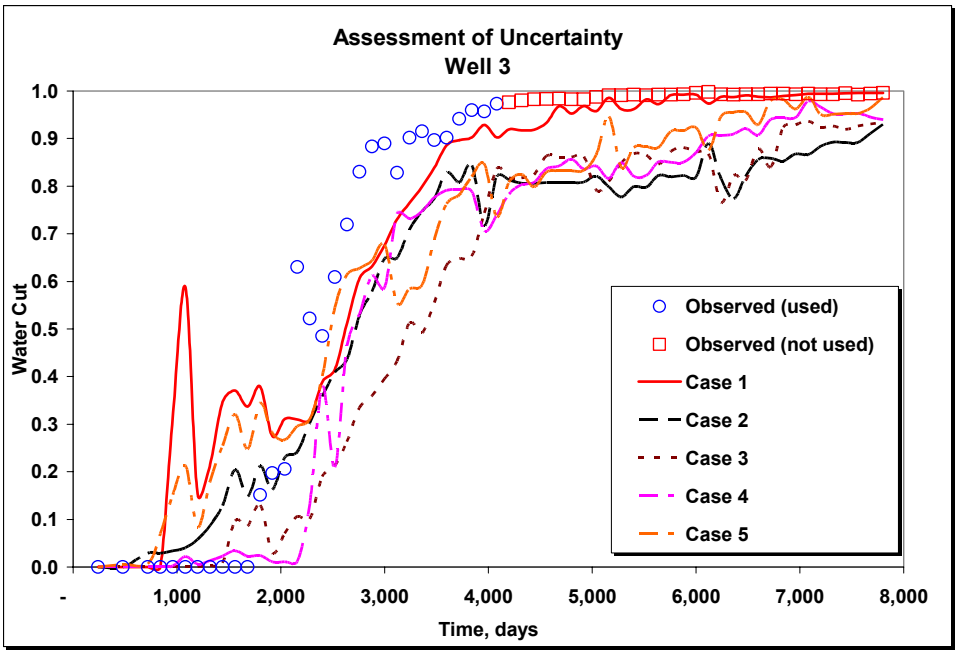
Because of the poor quality of the match, the predictions appear to be somewhat unsatisfactory. No conclusive remarks about uncertainty can be made until the matching is improved. This is an area of further investigation.



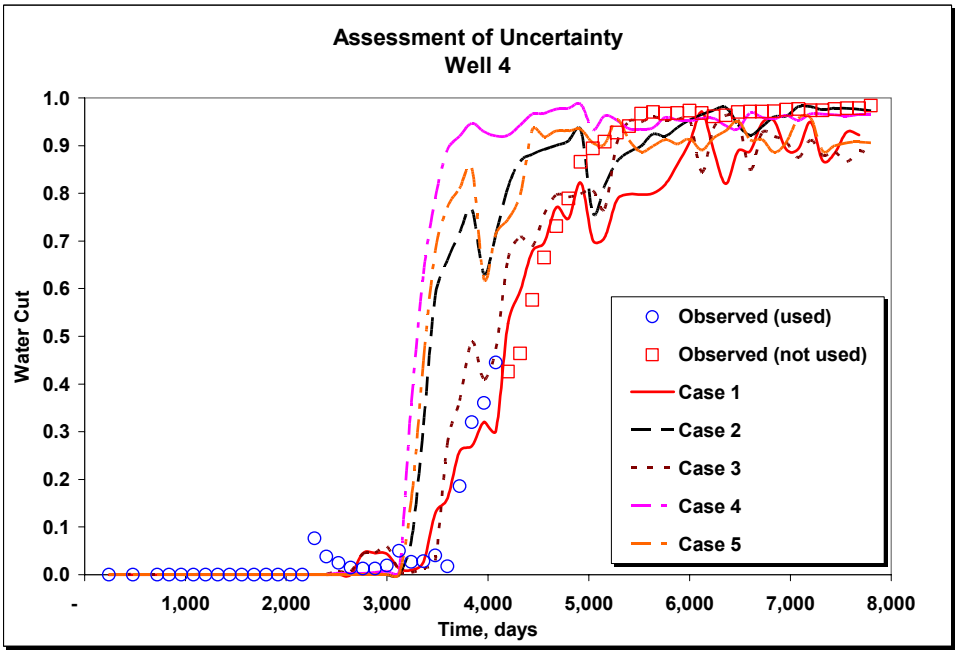
9.18 – Conditioned water cut using data up to 4,080 days, well 1. Prediction up to 7,800 days.



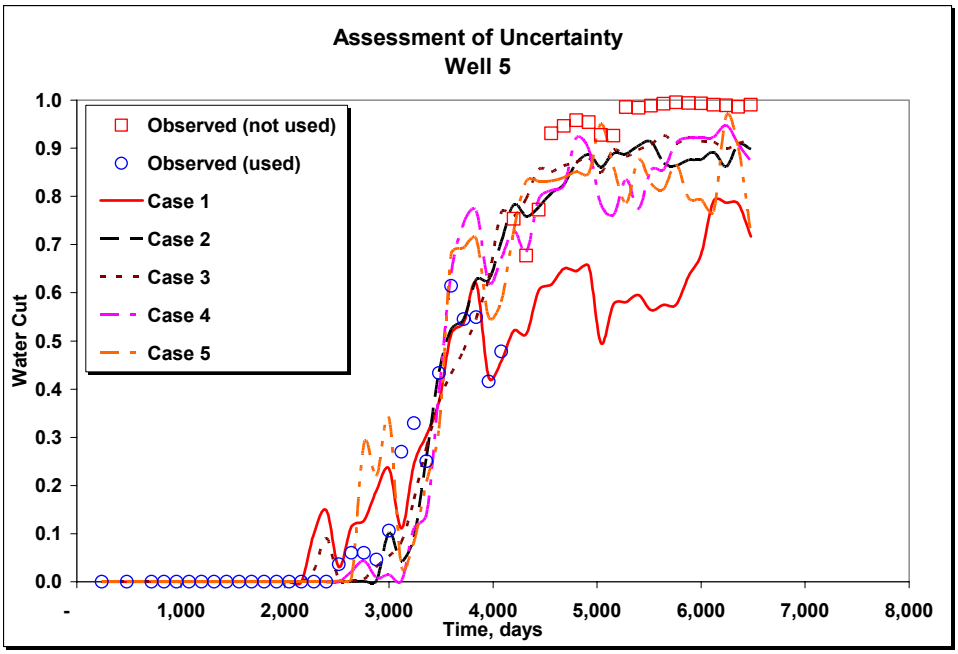
9.19 – Conditioned water cut using data up to 4,080 days, well 2. Prediction up to 7,800 days.



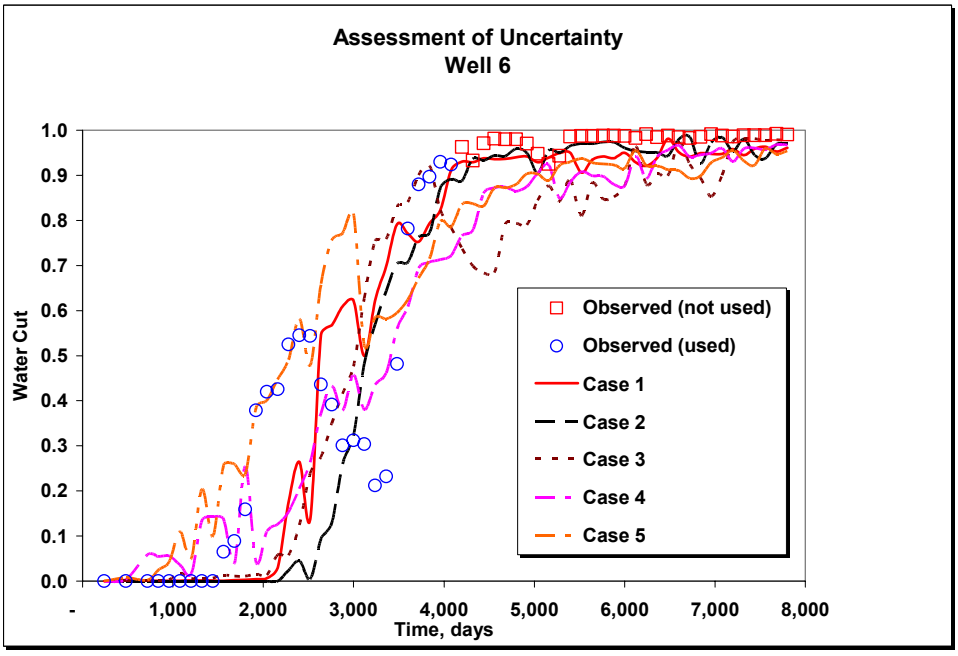
9.20 – Conditioned water cut using data up to 4,080 days, well 3. Prediction up to 7,800 days.



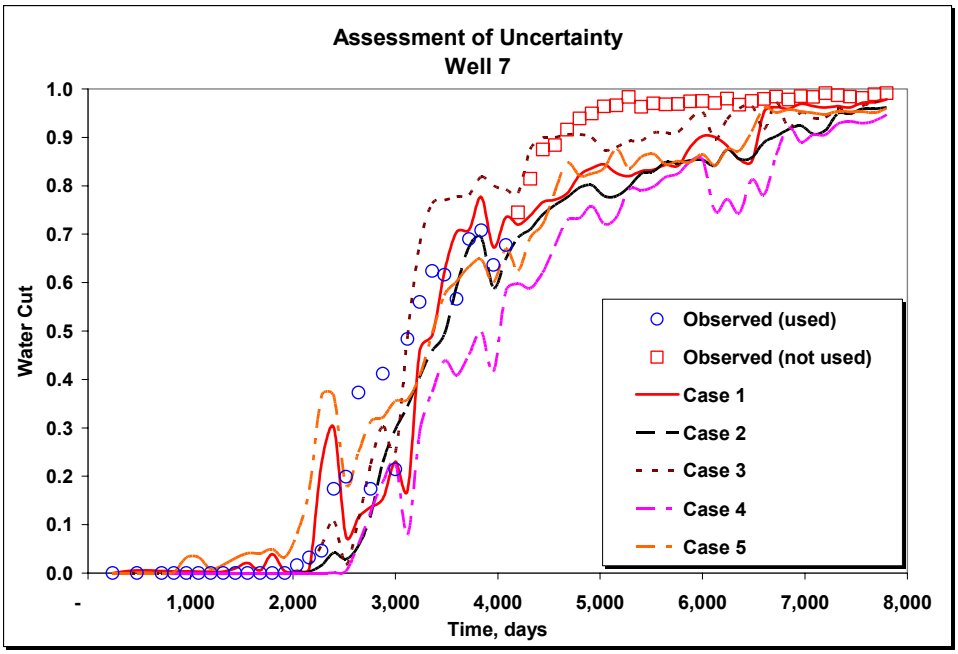
9.21 – Conditioned water cut using data up to 4,080 days, well 4. Prediction up to 7,800 days.



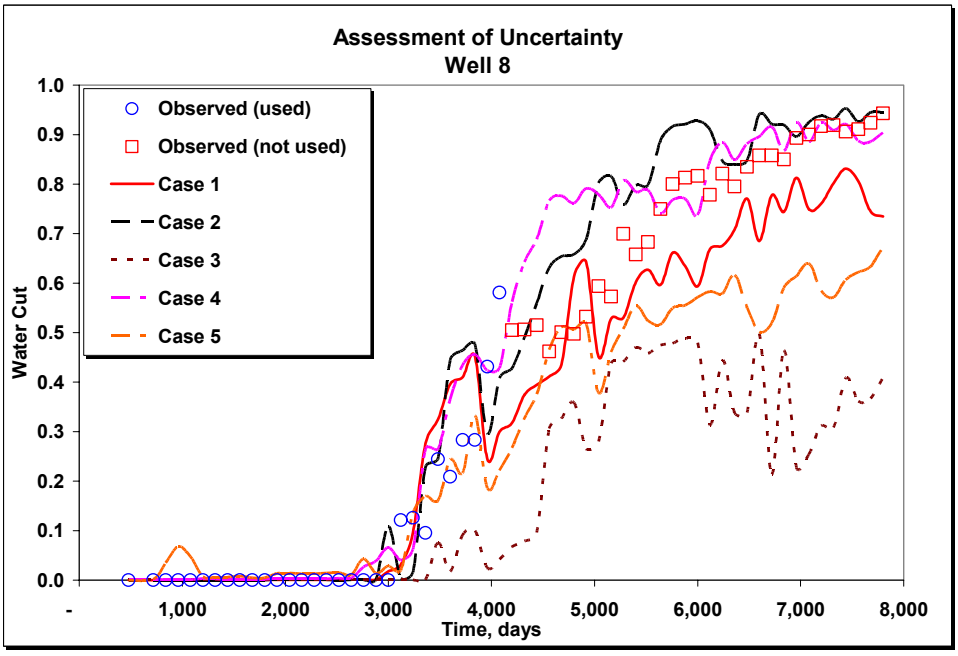
9.22 – Conditioned water cut using data up to 4,080 days, well 5. Prediction up to 7,800 days.



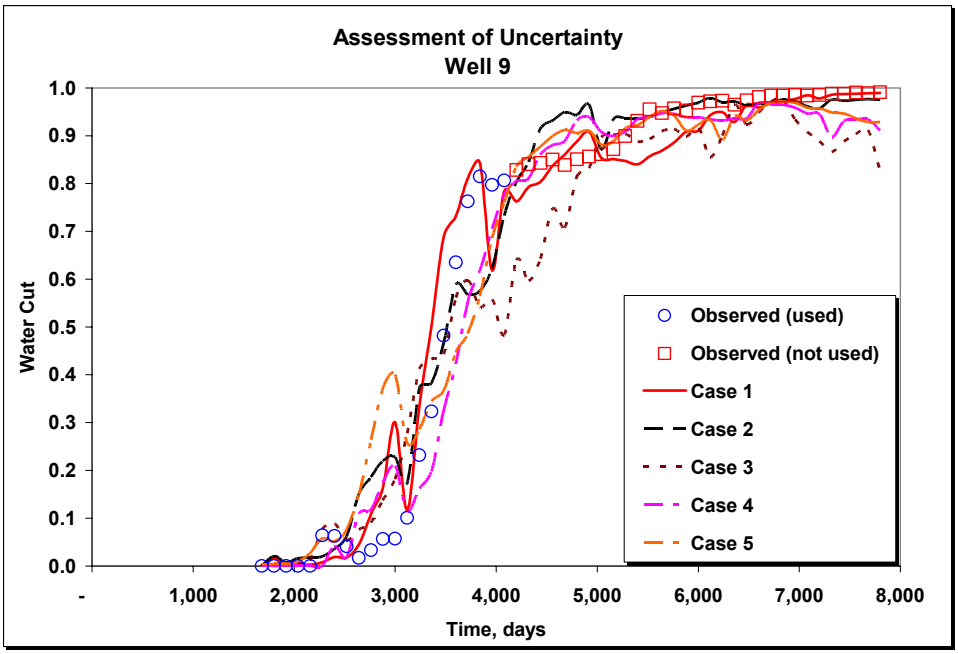
9.23 – Conditioned water cut using data up to 4,080 days, well 6. Prediction up to 7,800 days.



9.24 – Conditioned water cut using data up to 4,080 days, well 7. Prediction up to 7,800 days.



9.25 – Conditioned water cut using data up to 4,080 days, well 8. Prediction up to 7,800 days.



9.26 – Conditioned water cut using data up to 4,080 days, well 9. Prediction up to 7,800 days.

Summary

Field data from the Goldsmith field have been used to illustrate two efficient techniques:

1. Randomized Maximum Likelihood method. A technique to sample the ensemble of posterior realizations in which all transitions can be accepted.
2. An efficient Bayesian formulation. Its convenience is twofold:
    a. It accounts for the uncertainty in the prior model and the observed data by treating them probabilistically, and
    b. It is formulated in such a way that its objective function can be optimized using an efficient algorithm.

CHAPTER X

CONCLUSIONS AND RECOMMENDATIONS

Conclusions

In this dissertation, I have systematically investigated the computational scaling behavior of the Bayesian and the deterministic approaches to field–scale production data integration into high-resolution reservoir models. I also provide an efficient adaptation of the Bayesian approach that can lead to orders of magnitude savings in computation time for model sizes larger than 100,000 grid blocks. This is quite significant for practical applications considering that current reservoir models routinely consist of several hundred thousand to millions of grid blocks. I have also demonstrated a practical approach to uncertainty quantification during field-scale production data integration. Some specific conclusions from this study are as follows:

1. I have shown that the deterministic approach to inverse modeling utilizing the sparse matrix solver LSQR exhibits a linear scaling of computation time with respect to the number of grid blocks. In contrast, the Bayesian approach with the commonly used Gauss-Newton algorithm exhibits a quadratic scaling with respect to the number of grid blocks.
2. I have presented a fast and robust adaptation of the Bayesian formulation that preserves the statistical foundation of the Bayesian method and at the same time has a scaling property similar to that of the deterministic approach. This can lead to orders of magnitude savings in computation time for model sizes greater than 100,000 grid blocks.
3. My proposed method is based on an analytic computation of the square root of the inverse of the covariance matrix during production data integration using the Bayesian approach. We present a simple finite-difference stencil for the calculation of the square-root of the inverse. This allows us to pose the Bayesian

inverse problem in a manner analogous to the deterministic approach and the use of efficient sparse matrix solvers during the minimization of the data misfit.

4. I have demonstrated the power and validity of my method using synthetic examples. Finally, a field application from the Goldsmith field in west Texas demonstrates the practical feasibility of the approach.
5. The use of the new efficient Bayesian formulation along with the Randomized Maximum Likelihood method allows straightforward assessment of uncertainty. The former provides computational efficiency and the latter avoids rejection of expensive conditioned realizations.

Recommendations

It could be shown with the field application in Chapter IX that a quantitative measure of the change in uncertainty resulting from the integration of production data was feasible. However, this conclusion was based on the particular case when the water cut in all wells had risen to almost unity.

A more practical application would be to be able to condition the reservoir model using limited production data to then predict future reservoir performance. The scatter in the ensemble of predictive water cut curves would thus provide an uncertainty assessment tool.

REFERENCES

1. Bissell, R.C.: "Calculating Optimal Parameter for History Matching," *Proc*. 4th European Conference on the Mathematics of Oil Recovery, Topic E: History Match and Recovery Optimization, RØros, Norway (1994).
2. Oliver, D.S.: "Incorporation of Transient Pressure Data into Reservoir Characterization," *In Situ* (1994) **18,** 13.
3. Datta-Gupta, A., Vasco, D.W., Long, J.C.S., D'Onfro, P.S., and Rizer, W.D.: "Detailed Characterization of a Fractured Limestone Formation by Use of Stochastic Inverse Approach," *SPE Formation Evaluation* (Sept. 1995), **10,** 3.
4. Landa, J.L., Kamal, M.M., Jenkins, C.D., and Horne, R.N.: "Reservoir Characterization Constrained to Well Test Data: A Field Example," paper SPE 36511, presented at the 1996 SPE Annual Technical Conference and Exhibition, Denver, CO, 6-9 October.
5. Vasco, D.W., Datta-Gupta, A.: "Integrating Field Production History in Stochastic Reservoir Characterization," *SPE Formation Evaluation* (Sept. 1997), **12,** 3.
6. Reynolds, A.C., He, N., and Oliver, D.S.: "Reducing Uncertainty in Geostatistical Description with Well Testing Pressure Data," *Proc*., 1997 International Reservoir Characterization Conference, Houston, TX, 2-4 March.
7. Landa, J.L. and Horne, R.N.: "A Procedure to Integrate Well Test Data, Reservoir Performance History and 4-D Seismic Information into a Reservoir Description," paper SPE 38653 presented at the 1997 SPE Annual Technical Conference and Exhibition, San Antonio, TX, 5-8 October.
8. Wen, X., Deutsch, C. V., and Cullick, A. S.: "High Resolution Reservoir Models Integrating Multiple-Well Production Data," paper SPE 32728, presented at the 1997 SPE Annual Technical Conference and Exhibition, San Antonio, TX, 5-8 October, 1997.
9. Vasco, D. W., Yoon, S. and Datta-Gupta, A.: "Integrating Dynamic Data Into High Resolution Reservoir Models Using Streamline-Based Analytical Sensitivity Coefficients," *SPEJ*, (December 1999) **4**, 389.

10. Datta-Gupta, A., Nordas Kulkarni, K., Yoon, S., Vasco, D.W.: "Streamlines, Ray Tracing and Production Tomography: Generalization to Compressible Flow," *Petroleum Geoscience*, **7,** S75.
11. McLaughlin, D., Townley, L.R.: "A Reassessment of the Groundwater Inverse Problem," *Water Resources Research* (1996) **32,** 5.
12. Hadamard, J.: *Lecons sur la Propagation des Ondes et les Equations de l'Hydrodynamique*, Chelsea Publication Co., New York (1949).
13. Scales, J.A.: "To Bayes or not to Bayes," *Geophysics* (July-Aug. 1997) **62,** 1045.
14. Cobenas, R.H., Aprilian, S.S. Datta-Gupta, A.: "A Closer Look at Non-Uniqueness During Dynamic Data Integration into Reservoir Characterization," paper SPE 39669 presented at 1998 SPE/DOE Improved Oil Recovery Symposium, Tulsa, OK, 19-22 April.
15. Efron, B.: "Why Isn't Everyone a Bayesian," *American Statistician* (1989) **40,** 1.
16. Tikhonov, A.N., and Arsenin, V.Y.: *Solution of Ill-Posed Problems*, Halsted Press, New York (1977).
17. Paige, C.C., Saunders, M.A., "LSQR: An Algorithm for Sparse Linear Equations and Sparse Least Squares," *ACM Transactions on Mathematical Software* (March 1982) **8** (43).
18. Nolet, G.: "Seismic Wave Propagation and Seismic Tomography," *Seismic Tomography,* (1987) 1-23
19. Oliver, D.S.: "Calculation of the Inverse of the Covariance," *Mathematical Geology* (1998) **30**, 911.
20. Isaaks, E.H., Srivastava, R.M.: *An Introduction to Applied Geostatistics*, Oxford University Press, Oxford (1989)
21. Liu, N., Betancourt, S., Oliver, D.S.: "Assessment of Uncertainty Assessment Methods," paper SPE 71624, presented at the 2001 SPE ATCE, New Orleans, LA, 30 Sep-3 Oct.
22. Kitanidis, P.K.: "Quasi-linear Geostatistical Theory for Inversing," *Water Resour. Res*. (1995) **31**, 2411-2419

23. Oliver, D.S., He, N., Reynolds, A.C.: "Conditioning Permeability Fields to Pressure Data," 5th European Conference on the Mathematics of Oil Recovery, Leoben, Austria, 3-6 Sep, 1996
24. Wilks, S.S.: *Mathematical Statistics*, John Wiley and Sons, New York (1962)
25. Aly, A., Lee, W.J., Datta-Gupta, A., Mowafi, K., Prida, M., Latif, M.: "Application of Geostatistical Modeling in an Integrated Reservoir Simulation Study of the Lower Bahariya Reservoir, Egypt," SPE paper 53118 presented at the 1999 Middle East Oil Show and Conference , Bahrain, 20-23 Feb.
26. He, Z., Datta-Gupta, A.,Yoon, S.: "Streamline-based Production Data Integration Under Changing Field Conditions," SPE paper 71333 presented at the 2001 ATCE, New Orleans, LA, 30 Sep -3 Oct.

VITA

Leonardo Vega Velasquez
Petroleum Eng. Dept.
3116 TAMU
College Station, TX USA, 77843
Ph: (979)219-2325
Leovega61@yahoo.com

Education:

Doctor of Philosophy. Petroleum Engineering. Texas A&M University. December 2003

Master of Science. Petroleum Engineering. Texas A&M University. December 1998

Bachelor of Science. Petroleum Engineering. Universidad Nacional de Colombia. December 1986

Experience:

Summer Intern. ChevronTexaco. 2001

Summer Intern. Schlumberger. 2000

Texas A&M University. Research Assistant. 1997-2003

Ecopetrol. Reservoir Engineer. 1991-1995

Latest Publications:

Sinha, S., Kumar R., Vega, L., and Jalali, Y.: "Flow Equilibration Towards Horizontal Wells Using Downhole Valves," SPE paper 68635 presented at the 2001 SPE Asia Pacific Oil and Gas Conference and Exhibition, Jakarta, Indonesia, 17–19 April.

Vega, L., Datta-Gupta, A.: "Deterministic vs. Bayesian Approaches to Production Data Integration Into High-Resolution Reservoir Models," SPE paper 79666 presented at the 2003 17th Reservoir Simulation Symposium, Houston, Texas, 3-5 February.